

Приложение 1
Утверждено
Приказом Министерства экономики
Кыргызской Республики
от 25 декабря 2019 г. № 180

**МЕТОДИКА ПО ПРОГНОЗИРОВАНИЮ БЕДНОСТИ
С ИСПОЛЬЗОВАНИЕМ ПРОГРАММНОЙ СРЕДЫ «R»**

Содержание

Содержание	2
Список иллюстраций	3
Список таблиц	3
Список текстовых вставок	4
Список вставок из R.....	4
Сокращения	5
1 Введение.....	6
2 Прогнозирование бедности: основные понятия	7
2.1 Измерение бедности	8
2.2 Какую модель выбрать	8
2.3 Прогнозирование с логистической регрессией	11
3 Введение в данные	12
3.1 Интегрированное выборочное обследование бюджетов домашних хозяйств (ИОДХ) ..	12
3.2 Макроэкономические переменные на областном уровне	15
4 Стратегия построения моделей.....	16
4.1 1 Этап: Подготовка данных.....	16
4.2 2 Этап: Проверка данных	16
4.3 3 Этап: Оценка модели	16
4.4 4 Этап: Прогнозирование бедности.....	16
5 Подготовка данных	17
5.1 Импорт библиотек R для чтения данных.....	17
5.2 Чтение микроданных по домохозяйствам из НСК	17
5.3 Перекодирование потенциальных предикторов домохозяйств, предсказывающих статус бедности.....	19
5.4 Чтение региональных переменных и сопоставимость наборов данных	21
5.5 Выбор предсказывающих переменных	22
6 Проверка данных.....	25
6.1 Статистический тип переменных	25
6.2 Распределение численности.....	26
6.3 Отношения между бедными домохозяйствами и другими переменными	26
6.4 Использование весов выборки.....	27
7 Оценка модели.....	29
7.1 Включение в модель предсказывающих переменных.....	29
7.2 Оценка модели	31
7.3 Оценка роли каждой предсказывающей переменной.....	33
7.4 Оценка точности модели	37
7.4.1 Отклонение модели	37
7.4.2 Сгруппированные остатки.....	38
7.4.3 Вероятность ошибки.....	39
8 Прогнозирование бедности	41
8.1 Оценка уровня бедности в пределах выборки.....	41
8.2 Прогнозирование уровня бедности вне выборки.....	45
8.2.1 Прогнозирование бедности на 2017 г.	46

8.2.2	Прогнозирование бедности на 2018 г.	48
8.2.3	Прогнозирование бедности на 2019 г.	50
9	Приложения	52
9.1	Приложение 1: Начало работы в R / Rstudio	52
9.2	Приложение 2: Словарь специальных статистических терминов	55
9.3	Приложение 3: Слияние файлов	58
9.3.1	Дополнительные функции в R	59
9.3.2	Импорт файлов SPSS в R: «Библиотечное убежище» (library haven)	59
9.3.3	Импорт и обработка файла «бедность» - poverty (PROFIL_2016.sav)	60
9.3.4	Импорт и обработка других индивидуальных характеристик (f1_nal.sav)	61
9.3.5	Импорт и обработка характеристик жилищных условий (f7_01.sav)	62
9.3.6	Импорт и обработка наличия товаров длительного пользования (f7_02.sav)	63
9.3.7	Объединение наборов данных и сохранение данных	65
9.4	Приложение 4: Файлы кода R	67
9.4.1	Functions.R: дополнительные функции для создания первичного набора данных	67
9.4.2	data_2016_NSC.R: базовый скрипт для создания первичного набора данных для министерства экономики	68
9.4.3	data_2016_MoE.R: Подготовка данных	70
9.4.4	Checking_data_2016.R: Проверка данных	71
9.4.5	Model_estimation_2016.R: Оценка модели	72
9.4.6	Forecasting_model_2016.R: Прогнозирование бедности	73

Список иллюстраций

Рисунок 7-1:	Функция обратного логита $\text{logit}^{-1}(x)$: преобразование из линейных предикторов в вероятности, которое используется в логистической регрессии.	36
Рисунок 7-2:	График сгруппированных остатков для модели со взаимодействиями. Группы (бины) расположены неравномерно; скорее, каждый бин имеет одинаковое количество точек данных. Светлые линии на графике сгруппированных остатков показывают теоретические 95% -ные границы погрешности.	39
Рисунок 9-1:	Rstudio: панель по умолчанию	53
Рисунок 9-2:	Как открыть файл	54
Рисунок 9-3:	Как запустить строки в скрипт-файле	54
Рисунок 9-4:	Структура файлов ИОДХ 2016 г.	58

Список таблиц

Таблица 3-1	Структура вопросников ИОДХ 2016 г.	12
Таблица 5-1	Список переменных в окончательном наборе данных (data_forecasting_2016.RData)	22
Таблица 7-1	Список выбранных предсказывающих переменных, включенных в модель	30
Таблица 7-2:	Таблица перекрестной классификации	39

Таблица 8-1. Сравнение наблюдаемого и оценочного уровня бедности в 2016 году	44
--	----

Список текстовых вставок

Текстовая вставка 1-1. Как пользоваться этим руководством	7
Текстовая вставка 2-1. Отношение шансов	11
Текстовая вставка 3-1. Файл макроэкономических показателей в Excel	15
Текстовая вставка 7-1. Оценка по методу максимального правдоподобия	31
Текстовая вставка 7-2. Библиотека libraryarm	33
Текстовая вставка 7-3. Команда I()	33
Текстовая вставка 7-4. Член, характеризующий взаимодействие	33
Текстовая вставка 7-5. Правило $\beta/4$	36
Текстовая вставка 8-1. Прогноз ВВП на душу населения	46
Текстовая вставка 8-2. Разработки: корректировка весов выборки	47
Текстовая вставка 8-3. Другие возможные сценарии	48

Список вставок из R

Вставка из R 5-1. Как получать библиотеки в R	17
Вставка из R 5-2. Импорт файла уровня домохозяйства в R	18
Вставка из R 5-3. Перекодирование предсказывающих переменных и выходная переменная	20
Вставка из R 5-4. Импорт макроэкономических переменных	21
Вставка из R 5-5. Выбор предикторов для прогнозирования	22
Вставка из R 6-1. Распределение численности	26
Вставка из R 6-2. Перекрестная таблица	27
Вставка из R 6-3. Средневзвешенные значения	27
Вставка из R 6-4. Средневзвешенные значения по областям	28
Вставка из R 7-1. Оценка модели	32
Вставка из R 7-2. Оценённая модель: коэффициенты и стандартные ошибки	34
Вставка из R 7-3. Сгруппированные остатки (binnedresiduals)	38
Вставка из R 7-4. Коэффициент погрешности (errorrate)	40
Вставка из R 8-1. Импорт библиотек	41
Вставка из R 8-2. Загрузка данных	41
Вставка из R 8-3. Построение модели прогнозирования	41
Вставка из R 8-4. Прогнозирование в пределах выборки (In-sampleforecasting)	43
Вставка из R 8-5. Прогнозирование в пределах выборки по областям	43
Вставка из R 8-6. Сравнение наблюдаемого и предполагаемого уровня бедности в 2016 г.	44
Вставка из R 8-7. Средняя абсолютная ошибка (MAE)	45
Вставка из R 8-8. Импорт макроэкономического прогноза	45
Вставка из R 8-9. Обновление матрицы предикторов: 2017 год	46
Вставка из R 8-10. Прогноз вероятности быть бедным, 2017 год	47
Вставка из R 8-11. Прогноз уровня бедности на национальном уровне, 2017 год	47
Вставка из R 8-12. Прогноз уровня бедности на областном уровне, 2017 год	47
Вставка из R 8-13. Обновление матрицы предсказывающих переменных: 2018 год	48
Вставка из R 8-14. Прогноз вероятности быть бедным, 2018 год	49
Вставка из R 8-15. Прогноз уровня бедности на национальном уровне, 2018 год	49
Вставка из R 8-16. Прогноз уровня бедности на областном уровне, 2018 год	49
Вставка из R 8-17. Обновление матрицы предсказывающих переменных: 2019 год	50

Вставка из R 8-18. Прогноз вероятности быть бедным, 2019 год.....	50
Вставка из R 8-19. Прогноз уровня бедности на национальном уровне, 2019 год	50
Вставка из R 8-20. Экспорт результатов в файл Excel	51
Вставка из R 9-1. Импорт дополнительных функций	59
Вставка из R 9-2. Библиотечное убежище (library haven).....	60
Вставка из R 9-3. Импорт файла «бедность» - poverty	60
Вставка из R 9-4. Построение переменного количества членов домохозяйства по уровню образования.....	61
Вставка из R 9-5. Выбор характеристик домохозяйств	62
Вставка из R 9-6. Импорт и обработка файла жилищных условий	63
Вставка из R 9-7. Импорт и обработка файла наличия товаров длительного пользования	64
Вставка из R 9-8. Построение набора данных о наличии товаров длительного пользования на уровне домашних хозяйств.	65
Вставка из R 9-9. Слияние наборов данных	66

Сокращения

ВВП	(GDP) - Валовой внутренний продукт (используется взаимозаменяемо с ВРП)
ВРП	(GRP) - Валовой региональный продукт
KGS	- Кыргызский Сом
ИОДХ	- Интегрированное выборочное обследование бюджетов домашних хозяйств и рабочей силы в Кыргызской Республике (ИОДХ)
КР	- Кыргызская Республика
МЭ	- Министерство экономики Кыргызской Республики
НСК	- Национальный статистический комитет Кыргызской Республики

1 Введение

Кыргызская Республика привержена достижению целей устойчивого развития, в том числе Цели I - Ликвидация нищеты. Бедность и благосостояние являются первичными показателями социально-экономической ситуации в стране и оценкой политик, которые были использованы Правительством Кыргызской Республики. В тоже время решения государственной политики и подготовка правительственных мер требуют хорошего понимания факторов сокращения бедности и оценки их будущего воздействия на развитие бедности и благосостояния в стране. Прогнозирование является одним из необходимых инструментов для развития социально-экономической политики в правильном направлении и обеспечения финансовых ресурсов для принятия политических решений. Прогнозы бедности дополняют макроэкономические краткосрочные прогнозы (например, рост ВВП, занятость, инфляция) и способствует повышению важности вопросов распределения доходов между различными группами населения при оценке текущих экономических и социальных изменений (Navicke et al., 2014).

Данное операционное руководство описывает национальную методологию прогнозирования бедности в Кыргызской Республике для более обоснованных решений макроэкономической и социальной политики. Методология прогнозирования бедности тесно связана с существующей системой макроэкономического прогнозирования в Кыргызской Республике. Для выполнения данной задачи современная международная практика ориентирована на эконометрические модели для оценки влияния на индивидуальный риск бедности макроэкономических переменных, такие как «ВВП на душу населения» и «уровень безработицы» на страновом и региональном уровнях. Такие модели используют как макроэкономические данные, так и микроданные (на уровне домашних хозяйств). Построение комбинированных микро- и макро - моделей было признано подходящим подходом для анализа, в котором основное внимание уделяется влиянию макроэкономической политики и шоков на бедность (Bourguignon et al., 2008).

Целью настоящего руководства является предоставление подробных, поэтапных инструкций для оценки эконометрической модели прогнозирования бедности в Кыргызской Республике. Данное руководство предназначено для практиков и разработчиков госполитик, которые хотят проанализировать последствия воздействия государственных политик и макроэкономических прогнозов на бедность домашних хозяйств в Кыргызской Республике. В качестве источника данных в руководстве используется данные ИОДХ за 2016г., в которых объединены микро-данные и макро-данные. При этом это руководство предназначено для использования в последующие годы, оно написано в общем виде и, поэтому, применимо для любого года. Руководство написано для анализа бедности с использованием статистического программного пакета «R Studio» (версия 1.1.456 @ 2009-2018 RStudio, Inc.). Поэтому предоставленные коды и исследовательский материал предназначены для «R», и требуются некоторые базовые знания работы с «R».

В руководстве рассматриваются основы эконометрической теории наряду с некоторыми операционными вопросами и рекомендациями, связанными с выполнением моделирования прогнозов.

Руководство структурировано следующим образом:

В Разделе 2 даётся общий обзор стратегии прогнозирования бедности. Ключевым компонентом модели прогнозирования является наличие данных на уровне домохозяйств и данных на региональном / областном уровне. Описание данных представлено в Разделе 3. В Разделе 4 кратко описываются шаги, необходимые для построения модели прогнозирования. В последующих разделах подробно рассматривается каждый шаг. В частности, в Разделе 5 описывается, как читать

данные в R и как подготавливать данные. В Разделе 6 иллюстрируется, как проверять данные и как понять связь между характеристиками домохозяйства и бедностью. Этот раздел можно пропустить, если пользователь не хочет изменять выбор переменных, которые были включены в модель прогнозирования. В Разделе 7 описывается, как выполнить и оценить модель. В Разделе 8 показано, как получить прогнозы уровней бедности на национальном и областном уровне.

В Приложении 1 приведены пояснения о программном обеспечении R и среде RStudio для R, о том, как их установить, и проиллюстрированы первые основные этапы.

В Приложении 2 представлен основной глоссарий статистических терминов, используемых в руководстве.

Приложение 3 содержит подробную информацию о том, как импортировать файлы, объединять их и очищать наборы данных.

Приложение 4 показывает файлы синтаксиса в R.

Текстовая вставка1-1. Как пользоваться этим руководством

В этом руководстве пользователю представлены пошаговые инструкции с необходимыми инструментами для выполнения модели прогнозирования бедности с использованием бесплатного программного обеспечения «R» с открытым исходным кодом. Руководство содержит немного теории и практики. Следующие символы помогут вам в дальнейшем:



Этот символ просит вас обратить особое внимание на коды R.



Этот символ содержит статистические советы для анализа.



Этот символ указывает вам на дополнительную информацию и литературу.

2 Прогнозирование бедности: основные понятия

Валовой региональный продукт (ВРП) - показатель, измеряющий валовую добавленную стоимость, исчисляемый путём исключения из суммарной валовой продукции объёмов её промежуточного потребления.

Предиктор - прогностический параметр или предсказывающая переменная.

Детерминант - фактор, оказывающий влияние на конечный результат прогнозирования.

Прогнозирование - это предсказание будущего с максимально возможной точностью, с учётом всей доступной информации, включая исторические данные и знания о любых будущих событиях, которые могут повлиять на прогнозы (Гиндман и Афанасопулос / Hyndman and Athanasopoulos, 2018 г.).

Первое, что нужно установить, - это определение того, что прогнозировать. В прогнозировании бедности, прогнозируемой переменной является статус бедности домохозяйства (и его членов), с учётом его специфических характеристик, когда меняется экономический контекст региона, где живет домохозяйство.

В частности, когда некоторые региональные макропеременные, такие как валовой региональный продукт (ВРП) на душу населения и уровень безработицы, оказывают влияние с учётом существенных предикторов бедности на уровне домашних хозяйств. При контроле детерминантов бедности на уровне домохозяйств было бы полезно сравнить, насколько велико влияние региональных агрегатов по сравнению с такими микроуровневыми характеристиками.

- Как измеряется уровень бедности домохозяйства?
- Какую модель нужно принять?
- Как прогнозировать состояние бедности и уровень бедности?

2.1 Измерение бедности

Существует несколько определений и методов измерения бедности. Данное руководство относится к методологии измерения бедности, применяемой Национальным статистическим комитетом Кыргызской Республики (НСК).

Оценка бедности НСК основана на объективных измерениях потребления домашних хозяйств с определением черты бедности по основным потребностям. Основные потребности (т.е. минимальные потребности человека в пище, одежде и жилье) или абсолютная черта бедности - это оценённая в кыргызских сомах минимальная потребительская корзина, включающая компонент питания (черта продовольственной бедности) и компонент непродовольственных товаров и услуг. Черта продовольственной бедности - это оценочный уровень расходов, необходимых для получения 2100 калорий в день на человека, что считается минимальным необходимым уровнем ежедневного потребления продуктов питания для человека в среднем. Доля потребления непродовольственных товаров и услуг, соответственно, оценивается для тех домашних хозяйств, которые расположены чуть выше черты продовольственной бедности. Эта доля используется для оценки того, сколько можно потратить за непродовольственные товары. Оба расчёта (потребление продуктов питания и непродовольственных товаров и услуг) используются для определения общей черты бедности. Черта бедности определяется, по меньшей мере, раз в пять лет и корректируется с учётом инфляции в последующие годы.

Бедные домохозяйства определяются как те домохозяйства, потребление которых на душу населения ниже официальной черты бедности, установленной НСК. Все члены бедного домохозяйства считаются бедными. Это означает, что оценки бедности не учитывают потенциальное распределение ресурсов внутри домохозяйства. Абсолютный (или общий) уровень бедности - это процент населения, живущего в бедности.



Подробное описание измерения бедности в Кыргызстане содержится в Руководстве, подготовленном Всемирным банком и НСК, которое называется «Методологическое руководство: Измерение бедности в Кыргызской Республике»,

2.2 Какую модель выбрать

Для прогнозирования бедности требуется разработка эконометрической модели. Модель - это упрощенное представление процессов, которые происходят в реальном мире. Она должна быть репрезентативной в том смысле, что должна содержать характерные черты изучаемых явлений. В своей простейшей форме эконометрическая модель связана с изучением взаимосвязи между одной переменной, называемой **зависимой переменной (или переменной ответа или результата/** результирующей

переменной), и одной или несколькими переменными, называемыми **объясняющими переменными или предикторами (предсказывающими переменными)**. Для единообразия обозначений мы обозначим Y как зависимую переменную, а X_1, X_2, \dots, X_K - K предикторов.

Связь между Y и X устанавливается через функциональную форму f и набор параметров β следующим образом:

$$Y = f(X_1, X_2, \dots, X_K; \beta) + \text{error}$$

Ошибка – *error* или остаточный (возмущающий) член является случайной величиной, поскольку его значение нельзя контролировать или знать заранее. Ошибка представляет все те силы, помимо X , которые влияют на Y , но не могут быть явно введены в модель, а также чисто случайные силы.

Параметры $\beta = (\beta_0, \beta_1, \beta_2, \dots)$ являются неизвестными величинами, которые могут быть оценены на основе данных, которые, в широком смысле, представляют способ, которым объясняющие переменные связаны с зависимой переменной Y .

Когда Y - числовая непрерывная переменная, функциональная форма обычно линейна. Когда Y не является непрерывным, требуются другие функциональные формы.

Мы наблюдаем эти переменные на совокупности из n статистических единиц (называемых «популяция» или «население»). Общая единица обозначается i , $i = 1, 2, \dots, n$, а значения Y и X , измеренные для единицы i , обозначаются как $y_i, x_{1i}, \dots, x_{ki}, \dots, x_{Ki}$.

Определив функциональную форму и сделав некоторые допущения относительно ошибки *error*, можно оценить параметры по данным $y_i, x_{1i}, \dots, x_{ki}, \dots, x_{Ki}$, $i=1, 2, \dots, n$. Оценка модели по существу означает оценку параметров β .

Оценочные значения из модели являются ожидаемыми значениями зависимой переменной, если мы знали только значения предикторов. Если предположить, что функция f - линейная (модель линейной регрессии), это можно формализовать как:

$$\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_{1i} + \dots + \hat{\beta}_k x_{ki} + \dots + \hat{\beta}_K x_{Ki},$$

где «шапочка» означает «оценочный» или «прогнозный».

Целью модели, которую мы хотим выполнить, является оценка состояния бедности (с точки зрения вероятности быть бедным) каждого домохозяйства в стране на основе его характеристик и контекста, в котором живёт семья.

Зависимая переменная модели - это статус бедности. Это двоичная переменная, которая принимает значение 1, если домохозяйство бедное, и 0 - если домохозяйство не бедное, т.е. Y принимает значения

$$y_i = \begin{cases} 1 & \text{если домохозяйство } i - \text{бедное} \\ 0 & \text{если домохозяйство } i - \text{небедное} \end{cases}$$

Поскольку зависимая переменная является двоичной, мы не можем использовать линейную функциональную форму, но используем функцию «логит» - *logit*. Следовательно, эконометрическая модель, используемая для оценки уровня бедности домохозяйства, представляет собой так называемую **модель логистической регрессии - logistic regression model**.

Следующим шагом будет изучение того, как различные факторы (переменные) влияют на то, бедное это отдельно взятое домохозяйство или нет. Переменные могут быть такие как возраст, пол, размер домохозяйства и так далее. Эти переменные являются x -переменными в логистической регрессии, которые используются для объяснения значения переменной y .

Исследования и доступная литература по бедности показывают, что потенциальные объясняющие переменные (предикторы) бедности касаются социально-

экономических, демографических характеристик и характеристик человеческого капитала домохозяйства. Для любого конкретного домохозяйства вероятность его бедности определяется степенью и характером его участия на рынке труда. Это участие на рынке труда, в свою очередь, обусловлено характеристиками домохозяйства. Набор объясняющих переменных включает, например, род занятий (профессию) и статус занятости. Демографические переменные включают возраст (в зависимости от стадии жизненного цикла), пол главы домохозяйства и состав домохозяйства. Человеческий капитал домохозяйства описывается характеристиками образования. Переменными, которые определяют экономическую среду, в которой живут домохозяйства, могут быть валовой региональный продукт (ВРП) на душу населения, региональный уровень безработицы, сельскохозяйственное производство и т. д.

Модель логистической регрессии оценивает вероятность *probability* того, что Y примет значение 1, учитывая характеристики (то есть значение предикторов) статистических единиц i :

$$p_i = \Pr(y_i = 1 | \mathbf{x}) = \text{logit}^{-1}(\beta_0 + \beta_1 x_{1i} + \dots + \beta_k x_{ki} + \dots + \beta_K x_{Ki}) \quad (1)$$

где

p_i - это вероятность того, что домохозяйство i с характеристиками \mathbf{x}_i будет бедным;

β_0 - это отрезок, отсекаемый на координатной оси / «интерсепт» (Intercept);

$\beta_1, \beta_2, \dots, \beta_K$ - это коэффициенты, связанные с предикторами x_1, x_2, \dots, x_K соответственно.

Функция логита $\text{logit}^{-1}(z) = \frac{e^z}{1+e^z}$ в статистике известна как совокупная функция логистического распределения¹. Она преобразует непрерывные значения в диапазон (0, 1). Это необходимо, так как вероятности должны быть между 0 и 1: $0 < \Pr(y_i = 1 | \mathbf{x}_i) < 1$.

¹Она также известна как сигмовидная функция, поскольку она даёт S-образную кривую.

Текстовая вставка 2-1. Отношение шансов

Если $z_i = \beta_0 + \beta_1 x_{1i} + \dots + \beta_k x_{ki} + \dots + \beta_K x_{Ki}$, то $p_i = \frac{e^{z_i}}{1 + e^{z_i}}$.

Легко показать, что $(1 - p_i) = \frac{1}{1 + e^{z_i}}$.

Поэтому, мы можем записать $\frac{p_i}{1 - p_i} = e^{z_i}$.

Отношение $\frac{p_i}{1 - p_i}$ это просто соотношение шансов. Коэффициенты показывают отношение вероятности успеха к вероятности неудачи.

Если мы берем натуральный логарифм, мы получаем

$$\text{logit}(p_i) = \log\left(\frac{p_i}{1 - p_i}\right) = \beta_0 + \beta_1 x_{1i} + \dots + \beta_k x_{ki} + \dots + \beta_K x_{Ki}$$

то есть логарифм отношения шансов является линейным по X и по параметрам.

Логарифм $\log\left(\frac{p_i}{1 - p_i}\right)$ называется логит - logit, а название модели - logitmodel.

В более краткой форме - это:

$$\text{logit}(p_i) = \log\left(\frac{p_i}{1 - p_i}\right) = x_i' \beta.$$

Поэтому в уравнении (1) мы используем символ logit^{-1} .



Рекомендуемая литература:

Hyndman R.J. and G. Athanasopoulos (2018), Forecasting. Principles and Practice, Second Edition, OTexts. Also available online at <https://otexts.com/fpp2/>.

Gujarati D.N. (2006), Essentials of Econometrics, Third Edition, McGraw-Hill.

Gelman A. and J. Hill (2006), Data Analysis Using Regression and Multilevel Models/Hierarchical Models, Cambridge University Press.

Agresti A. (2018), Statistical Methods for the Social Sciences, Fifth Edition, Pearson.

2.3 Прогнозирование с логистической регрессией

Как только модель была оценена с использованием уже существующих данных, прогноз вероятности быть бедным для каждого домашнего хозяйства в выборке определяется путём изменения микропеременных (характеристики домашнего хозяйства) и макропеременных (социально-экономические условия, в которых живёт домохозяйство).

Национальные и региональные уровни бедности затем прогнозируются как взвешенная сумма вероятностей прогноза на уровне домохозяйства, где веса - это весовые коэффициенты выборки обследования. Весовые коэффициенты выборки, на уровне домохозяйства, также могут меняться. С течением времени модель можно адаптировать к изменениям характеристик населения через новый набор весов выборки для того, чтобы соответствовать прогнозу по населению (например, по мере того, как прогнозируется изменение размеров домохозяйств, то новые весовые коэффициенты будут включать воздействие от такого изменения на состав домохозяйств).

3 Введение в данные

3.1 Интегрированное выборочное обследование бюджетов домашних хозяйств (ИОДХ)

Интегрированное выборочное обследование бюджетов домашних хозяйств и рабочей силы в Кыргызской Республике (ИОДХ) было представлено в 2003 году Национальным статистическим комитетом (НСК) и охватывает примерно 5000 домашних хозяйств ежеквартально в соответствии с Постановлением Правительства Кыргызской Республики «О статистике выборочных обследований домашних хозяйств» от 17 января 2004 года № 25 и № 281 (от 10 июня 2008 года).

Выборка домохозяйств для ИОДХ осуществляется с использованием метода стратифицированной двухступенчатой случайной выборки. Страна разделена на 16 страт / слоёв, представляющих собой городские и сельские домохозяйства в семи областях (Баткенская, Джалал-Абадская, Иссык-Кульская, Нарынская, Ошская, Таласская и Чуйская), городе Бишкек и городе Ош. На первом этапе выборки выбирается сельские и городские районы области, а на втором этапе из каждой страты делается выборка домохозяйств. Домашние хозяйства в выборке связаны с весами выборки. Весовые коэффициенты выборки позволяют на основании данных по домашним хозяйствам, включённым в выборку, анализировать население, из которого они были выбраны. В ИОДХ доступны веса, как по домохозяйствам, так и по населению. Размер выборки обеспечивает надёжные оценки бедности на национальном, городском, сельском и областном уровнях.

В ИОДХ используются опросники в семи формах, каждая форма имеет различное назначение и периодичность. Структура анкеты (на 2016 год) показана в следующей таблице.

Таблица 3-1 Структура вопросников ИОДХ 2016 г.

	Форма	Периодичность сбора данных	Единица обследования	Разделы
Ф1	Контрольная карточка домашнего хозяйства	Квартально	Индивидуальная	Список лиц д/х
Ф2	Социально – демографические характеристики	Ежегодно	Индивидуальная	1. Образование 2. Миграция 3. Состояние здоровья 4. Женская анкета
Ф3	Расходы на продукты питания	Квартально(Двухнедельные дневники)	Домохозяйство	1. Покупка продуктов питания 2. Потребление продуктов питания 3. Расходы на питание вне дома
Ф4	Занятость и безработица	Квартально	Индивидуальная (от 15 лет)	LFS
Ф5	Расходы на непродовольственные товары	Квартально	Домохозяйство	Покупка непродовольственных товаров

Ф6	Доходы расходы	и	Квартально	Домохозяйство	1. Некоторые непродовольственные товары 2. Жилищно-коммунальные расходы 3. Расходы на здравоохранение 4. Расходы на транспорт 5. Расходы на образование и уход за детьми 6. Прочие расходы семьи 7. ЛПХ и скот 8. Доход д/х
Ф7	Личное хозяйство жилищные условия	и	Ежегодно	Домохозяйство	1. Жилищные условия 2. Предметы длительного пользования 3. Личное подсобное хозяйство

Данные опроса записаны в двух различных типах файлов SPSS (*.sav): исходных данных из разделов опросников и вторичный файл бедности (PROFIL_2016.sav), разработанный НСК. Файлы первичных данных организованы по формам и разделам.

Например, файл f3_01.sav содержит данные из раздела 1 формы 3 (Покупка продуктов питания), а файл f7_02.sav содержит данные из Раздела 7.2 исходного вопросника. В больших формах файлы данных ещё больше разбиваются, в соответствии с подразделами вопросника.

Кроме того, НСК предоставляет один вторичный файл, PROFIL_2016.sav, который является основным файлом анализа. В этом файле содержится подробная информация о расходах домохозяйства, потреблении, доходах и некоторых социально-демографических характеристиках домохозяйства и главы домохозяйства, а также черты бедности и показатели бедности. Переменные стратификации (расслоения) и весов обследования находятся в файле Basic.sav. Другая соответствующая информация (например, информация об уровне образования всех членов домохозяйства) доступна в необработанном файле f1_nal.sav.

В рамках этой структуры микроданные, важные для прогнозирования, будут представлены в виде файла .sav, полученного путем объединения на уровне домохозяйств соответствующей информации из файла PROFIL_2016.sav, f1_nal.sav, f7_01.sav (жилищные условия) и из файла f7_02.sav (наличие товаров длительного пользования).

Подробная информация о том, как объединить эти файлы в R, представлена в Приложении 3.

Объединенный файл, который называется KINS_2016.sav содержит 4889 строк, соответствующих 4889 домохозяйствам, опрошенных в обследовании 2016 года, и 46 переменных, как указано ниже.

Таблица 3–2 Список переменных в исходном файле микроданных (KINS_2016.sav)

Название	Описание
hh code	идентификационный код д/х
weight	вес выборки (уровень населения)
expfact	вес выборки (уровень домохозяйства)

year	год проведения анализа
inc2	денежные переводы (тыс. сом)
oblast	область проживания
b002	город (1) или село (2)
hhhage	возраст главы д/х
hhheduc	образование главы д/х
hhhempl	статус занятости главы д/х
hhsex	пол главы д/х
hsize	количество членов д/х
Nempl	кол-во работающих членов д/х
NHighEduc	кол-во членов д/х с высшим образованием
NProfEduc	кол-во членов д/х с профессиональным образ-ем
pccd	потребление д/х на душу населения с дефлятором (выражен в постоянных ценах)
pline	черта бедности
cpsec	статус бедности д/х
type_accommodation	тип жилища
housing_ownership	форма собственности дома
getting_accommodation	как вы получили это жилье
total_area_sm	общая площадь, занимаемая семьей (кв. м)
living_area_sm	жилая площадь, занимаемая семьей (кв. м)
number_of_rooms	кол-во жилых комнат в жилище
walls_material	основной материал стен жилища
centralized_heating	наличие централизованного отопления
individual_heating_system	наличие индивидуального отопления
water_pipes	наличие водопроводных труб
sewage	наличие канализации
hot_water_supply	наличие горячего водоснабжения
centralized_gas_supply	наличие централизованного газоснабжения
bath_or_shower	наличие ванны или душа
telephone	наличие телефона
electric_stove	наличие электроплиты (стационарная)
electricity	наличие электричества
SatAntenna	наличие спутниковой антенны в д/х
ElOven	наличие электрической духовки
TVCol	наличие цветного телевизора
LandLine	наличие стационарного телефона
PC	наличие ПК
Car	наличие автомобиля
WshMachine	наличие стиральной машины
Sofa	наличие дивана
Ref1	наличие 1-камерного холодильника
Ref2	наличие 2/3-х камерного холодильника
Freezer	наличие морозильной камеры



Для получения дополнительной информации см.: <http://www.stat.kg/>



Перед использованием набора данных другого года важно убедиться, что структура файла такая же, как и в предыдущем году. Также проверьте анкеты.

Были ли добавлены дополнительные вопросы или разделы? Были ли перенумерованы или удалены вопросы?

3.2 Макроэкономические переменные на областном уровне

База данных со всей информацией на уровне домохозяйства должна быть дополнена соответствующей информацией на областном уровне. Необходимо импортировать набор данных региональных макроэкономических показателей.

Региональные индикаторы, представленные в виде файла Excel, располагаются в таблице из 9 строк (количество областей плюс города Бишкек и Ош) и столбцов по количеству индикаторов. Региональные показатели должны быть по текущему году (в данном случае, 2016 год), а также для тех лет, по которым мы хотим прогнозировать уровень бедности (в данном случае, 2017 - 2021 гг.). Следующая Текстовая вставка дает представление о том, как должен выглядеть файл Excel.

Текстовая вставка 3-1. Файл макроэкономических показателей в Excel

Региональные переменные в Excel

Файл Excel с региональными переменными, который называется `Oblast_dataset.xls`, содержит текущие и прогнозируемые значения ВВП на душу населения и уровня безработицы на областном уровне, рассчитанные Министерством экономики.

Это должно быть организовано следующим образом.

Oblast	GDP_2016	GDP_2017	GDP_2018	GDP_2019	GDP_2020	GDP_2021	Unemp_2016	Unemp_2017	Unemp_2018	Unemp_2019	Unemp_2020	Unemp_2021
41702 Issykul	131,6	138,0	131,1	134,8	141,8	134,1	8,9	8,8	8,7	8,5	8,4	8,3
41703 Jalal-Abad	48,4	55,2	55,3	56,6	56,8	56,1	7,5	7,4	7,3	7,1	7,0	6,9
41704 Naryn	53,1	51,7	51,6	53,5	53,1	51,4	9,0	8,9	8,8	8,6	8,5	8,4
41705 Batken	37,9	38,9	38,1	39,2	39,4	38,6	10,2	10,1	10,0	9,8	9,7	9,6
41706 Osh	28,0	31,1	30,7	31,5	31,5	31,1	5,5	5,4	5,3	5,1	5,0	4,9
41707 Talas	59,8	63,9	64,5	65,8	65,5	64,5	2,8	2,7	2,6	2,4	2,3	2,2
41708 Chui	88,7	98,2	98,5	102,6	102,0	99,8	8,4	8,3	8,2	8,0	7,9	7,8
41711 Bishkek	181,0	196,8	196,4	201,7	201,6	198,4	7,7	7,6	7,5	7,3	7,2	7,1
41721 Osh city	95,3	112,5	112,6	115,0	117,6	115,6	3,6	3,5	3,4	3,2	3,1	3,0

4 Стратегия построения моделей

В этом разделе кратко описаны шаги по созданию модели для прогнозирования бедности. В следующих разделах будет подробно рассмотрен каждый шаг.

4.1 1 Этап: Подготовка данных

Этот этап касается всех процедур, необходимых для подготовки набора данных, который будет использоваться в качестве входных данных для модели прогнозирования:

- Импорт библиотек R для чтения данных в SPSS и / или Excel, представленных НСК или МЭ;
- Считывание микроданных по домохозяйствам из НСК
- Перекодировка потенциальных предикторов домохозяйств, предсказывающих статус бедности
- Считывание региональных переменных, полученных из Министерства экономики и приведение наборов данных к сопоставлению
- Выбор предикторов (предсказывающих переменных) и сохранение файла

4.2 2 Этап: Проверка данных

Этот этап касается описания основных переменных, записанных в наборе данных. Этот этап важен, чтобы узнать и быть уверенным в предсказывающих статус бедности переменных.

- Понимание статистического типа переменных
- Частота бедных домохозяйств по областям с использованием весов выборки
- Взвешенный процент бедных домохозяйств по областям с использованием весов выборки
- Условные средние значения предсказывающих переменных по статусу бедности

4.3 3 Этап: Оценка модели

Этот этап относится к оценке, проверке и использованию модели для прогнозирования:

- Включение предикторных переменных в модель в соответствии с их статистическим типом
- Оценка модели
- Оценка роли каждого предиктора
- Проверка и точность модели

4.4 4 Этап: Прогнозирование бедности

Этот последний этап позволяет пользователю спрогнозировать состояние бедности на основе модели, выбранной на 3 Этапе.

- Построение модели прогнозирования на основе оценки модели, как в 3 Этапе
- Прогнозирование уровня бедности внутри выборки
- Обновление переменных прогноза (предсказывающих переменных)
- Прогнозирование уровня бедности вне выборки
- Прогнозирование уровней бедности на национальном и региональном уровнях.

5 Подготовка данных

5.1 Импорт библиотек R для чтения данных

Как указано ниже, микроданные поступают из Национального статистического комитета в виде файла SPSS, а макроданные поступают из Министерства экономики в виде файла Excel. Поэтому нам нужны две библиотеки в R для импорта файлов *.sav и *.xls.

Библиотеки R - это коллекции функций и наборов данных, разработанных сообществом. Они увеличивают мощность R, улучшая существующие базовые функции R или добавляя новые. Для установки библиотек необходимо подключение к зеркалам CRAN.

Для этого необходимо запустить следующие строки в R, которые позволяют переходить непосредственно к зеркалам CRAN и, во всей функции, устанавливать две необходимые библиотеки: библиотека `haven` для файлов SPSS и библиотека `rio` для файлов Excel.

Вставка из R5-1. Как получать библиотеки в R

```
#=====
#CRAN package repository
#=====
options(repos =c(CRAN ="http://cran.rstudio.com"))
#=====
#Getting the libraries
#=====
# requiring packages
ipak <-function(pkg){
  new.pkg <-pkg[!(pkg %in%installed.packages()[, "Package"])]
  if (length(new.pkg))
    install.packages(new.pkg, dependencies =TRUE)
  sapply(pkg, require, character.only =TRUE)
} ##end of requiring packages function

packages <-c("haven", "rio")
ipak(packages)
```

Выходные данные R должны показать значение«TRUE», TRUE указывает, что библиотеки установлены правильно:

```
## Loading required package: haven
## Loading required package: rio
## haven    rio
## TRUE     TRUE
```

5.2 Чтение микроданных по домохозяйствам из НСК

После того, как библиотеки установлены, для считывания микроданных необходимо установить каталог, в котором находятся данные, и перейти к считыванию файла. Просмотр данных и проверка на отсутствие данных может помочь понять, правильно ли импортированы данные.

Ниже показаны строки в R для установления каталога, считывания данных и проверки, содержит ли переменная недостающую информацию и сколько их:

Вставка из R5-2. Импорт файла уровня домохозяйства в R

```
#####
#The household-level dataset
#####
## Setting the working directory
setwd("C:/YOUR DIRECTORY")
## Reading the file in SPSS---as given from NSC
KIHS_2016<-as.data.frame(read_sav("KIHS_2016.sav"))
KIHS_2016[1:4,]

## Checking for missing data
sumNA<-function(vec){
sum(is.na(vec))
}
apply(KIHS_2016, 2, sumNA)
```

Если данные импортированы должным образом, на выходе будет следующее:

```
## hh_code weight expfact year inc2 oblast b002 hhhage hheduc hhempl
## 1 20001 893.1740 297.7247 2016 0 41702 2 46 5 1
## 2 20002 314.0635 314.0635 2016 0 41702 2 41 5 1
## 3 20003 1463.5455 292.7091 2016 0 41702 2 45 5 1
## 4 20004 414.5229 207.2614 2016 0 41702 2 66 8 0
## hhhsex hsize Nempl NHighEduc NProfEduc pccd pline cpssc
## 1 2 3 1 0 0 50178.56 31145.45 0
## 2 1 1 1 0 0 75264.72 31145.45 0
## 3 2 5 3 0 0 56405.15 31145.45 0
## 4 1 2 0 0 0 42813.31 31145.45 0
## type_accommodation housing_ownership getting_accommodation total_area_sm
## 1 4 3 2 34.6
## 2 3 3 2 54.0
## 3 3 3 5 82.0
## 4 3 3 2 61.5
## living_area_sm number_of_rooms walls_material centralized_heating
## 1 26.2 2 1 2
## 2 36.0 2 3 2
## 3 61.5 4 4 2
## 4 34.0 3 3 2
## individual_heating_system water_pipes sewage hot_water_supply
## 1 1 1 1 2
## 2 1 1 1 2
## 3 1 1 1 2
## 4 1 1 1 2
## centralized_gas_supply bath_or_shower telephone electric_stove
## 1 2 2 2 2
## 2 2 2 2 2
## 3 2 2 2 2
## 4 2 2 2 2
## electricity SatAntenna ElOven TVCol LandLine PC Car WshMachine Sofa Ref1
## 1 1 1 1 0 0 0 0 1 0
## 2 1 0 0 1 0 0 0 0 1
## 3 1 0 0 1 0 0 1 0 0
## 4 1 1 0 1 0 0 0 0 0 1
## Ref2 Freezer
## 1 0 0
## 2 0 0
## 3 0 0
## 4 0 0
```

```

##          hh_code          weight
##          0              0
##      expfact          year
##          0              0
##          inc2          oblast
##          0              0
##          b002          hhhage
##          0              0
##          hheduc          hhempl
##          0              0
##          hhhsex          hsize
##          0              0
##          Nempl          NHighEduc
##          0              0
##          NProfEduc      pccd
##          0              0
##          pline          cpsc
##          0              0
##      type_accommodation  housing_ownership
##          0              0
##      getting_accommodation  total_area_sm
##          0              0
##          living_area_sm      number_of_rooms
##          0              0
##          walls_material      centralized_heating
##          0              0
##      individual_heating_system  water_pipes
##          0              0
##          sewage          hot_water_supply
##          0              0
##      centralized_gas_supply      bath_or_shower
##          0              0
##          telephone          electric_stove
##          0              0
##          electricity          SatAntenna
##          0              0
##          ElOven          TVCol
##          0              0
##          LandLine          PC
##          0              0
##          Car          WshMachine
##          0              0
##          Sofa          Ref1
##          0              0
##          Ref2          Freezer
##          0              0

```

В файле не должно быть пропущенных данных. Когда числа отличаются от нуля, соответствующие переменные имеют недостающую информацию. Отсутствующие значения в этом наборе данных указывают на проблемы при чтении данных, поскольку предполагается, что набор данных должен быть полным.

5.3 Перекодирование потенциальных предикторов домохозяйств, предсказывающих статус бедности

Перекодирование (Recoding) - очень важный и предварительный шаг в анализе. Могут быть ситуации, в которых переменные представляют категории / классы,

которые являются пустыми или имеют только несколько наблюдений, или ситуации, в которых мы гораздо больше заинтересованы в одних категориях, чем в других.

Необходимо посмотреть на распределение переменной, чтобы увидеть, может её лучше перекодировать, избегая категорий с малым количеством записей, и определить, есть ли смысл объединить некоторые из этих категорий, свернув существующую переменную в меньшее количество категорий, содержащих разумное количество наблюдений. Например, в исходном наборе данных переменная `hhheduc`, соответствующая уровню образования главы домохозяйства, имеет много категорий (Высшее образование, Неполное высшее, Среднее специальное, Профессиональное техническое, Общее среднее (полное), Общее среднее (неполное), Начальное, Нет начального, Неграмотный), и некоторые из них имеют только несколько записей: 31 человек в категории 2 (неполное высшее образование), 33 человека в категории 8 (нет начальной школы) и 37 человек в категории 9 (неграмотный). В таких ситуациях всегда настоятельно рекомендуется свернуть исходные категории в меньшее количество классов, чтобы иметь переменную с несколькими классами / категориями, но с разумным количеством записей в каждом классе.

Иногда также очень полезно перекодировать переменную для лучшей интерпретации и обеспечения согласованности между переменными. Например, значение 1 или 0 должно иметь одинаковое значение для всех переменных.

Команда в R для перекодирования переменной - `ifelse`. В этом анализе, используя `ifelse`, некоторые переменные перекодируются следующим образом.

Вставка из R5-3.Перекодирование предсказывающих переменных и выходная переменная

```
#=====
# New recodification of potential predictors and outcome
#=====
## Area of residence
KIHS_2016$Area<-ifelse(KIHS_2016$b002==1, "Urban", "Rural")

##Type of accommodation
KIHS_2016$Rooms<-ifelse(KIHS_2016$number_of_rooms>=6, 6,
KIHS_2016$number_of_rooms)

##mq of the toral are of the housing
KIHS_2016$Mq<-ifelse(KIHS_2016$total_area_sm>=300, 300, KIHS_2016$total_area_sm)

##Availability of sewer services
KIHS_2016$Sewer<-ifelse(KIHS_2016$sewage==1, 1, 0)

##Availability of electric stove
KIHS_2016$ElStove<-ifelse(KIHS_2016$electric_stove==1, 1, 0)

##Size of household's size
KIHS_2016$Hsize<-ifelse(KIHS_2016$hsize>=7, 7, KIHS_2016$hsize)

##Household's head gender
KIHS_2016$hhhsex<-ifelse(KIHS_2016$hhhsex==1, "1 Male", "2 Female")

##Household's head education
KIHS_2016$Educ4<-ifelse(KIHS_2016$hhheduc==1,"Higher Educ",
ifelse (KIHS_2016$hhheduc==2|
KIHS_2016$hhheduc==3|
KIHS_2016$hhheduc==4|
KIHS_2016$hhheduc==41|
KIHS_2016$hhheduc==42, "Prof. Educ",
"Secondary or less"))
```

```

KIHS_2016$Educ4<-relevel(as.factor(KIHS_2016$Educ4), ref ="Secondary or less")

##Household's head employment Status
KIHS_2016$HHempl<-ifelse(KIHS_2016$hhempl==1, 1, 0)

##Number of employed in the HH
KIHS_2016$Nempl<-ifelse(KIHS_2016$Nempl>3, 4, KIHS_2016$Nempl)

##Giving names to oblasts
KIHS_2016$oblast<-as.factor(ifelse(KIHS_2016$oblast==41702, "41702 Issykul",
ifelse(KIHS_2016$oblast==41703, "41703 Jalal-Abad",
ifelse(KIHS_2016$oblast==41704, "41704 Naryn",
ifelse(KIHS_2016$oblast==41705, "41705 Batken",
ifelse(KIHS_2016$oblast==41706, "41706 Osh",
ifelse(KIHS_2016$oblast==41707, "41707 Talas",
ifelse(KIHS_2016$oblast==41708, "41708 Chui",
ifelse(KIHS_2016$oblast==41711, "41711 Bishkek",
"41721 Osh city"))))))))

## Outcome variable
KIHS_2016$poor<-ifelse(KIHS_2016$cpssc==100, 1, 0)

```



R имеет отличную функцию помощи. В окне консоли RStudio введите `?ifelse`, и появится окно с подробной информацией о команде и способах ее использования.

5.4 Чтение региональных переменных и сопоставимость наборов данных

Переменные регионального уровня находятся в файле Excel. Поэтому, для чтения файла в R, требуется библиотека `rio`. Эта библиотека уже установлена и готова к использованию (см. Раздел 5.1). Как обычно, для чтения данных необходимо установить каталог, в котором находятся данные, а потом считать файл (команда R - `import`). Импортировав набор данных, необходимо создать переменную на микроуровне, где каждое домохозяйство должно иметь одинаковое значение региональной экономической переменной, в которой живет домохозяйство.

В этом анализе две новые переменные добавляются к существующему набору данных `KIHS_2016`: одна относится к валовому региональному продукту на душу населения, а другая - к региональному уровню безработицы. Следующие строки R - это считывание файла Excel и создание новых переменных на уровне домохозяйства на основе региональных данных.

Вставка из R5-4. Импорт макроэкономических переменных

```

#=====
#The Oblast-level dataset
#=====
all.oblast <-import("C:/YOUR DIRECTORY/Oblast_dataset.xls")
KIHS_2016$full.GDP <-all.oblast$GDP_2016[KIHS_2016$oblast]
## Per capita GDP at oblast level
KIHS_2016$full.UNEMP <-all.oblast$Unemp_2016[KIHS_2016$oblast]

```

```
## Unemployment rateat oblast level
```

Набор данных KHS_2016 теперь содержит 4889 строки 48 (46 + 2) столбцов.

5.5 Выбор предсказывающих переменных

После того, как набор данных на микро- (домохозяйство) и макро- (область) уровнях был согласован, рекомендуется оставить в наборе данных только те переменные, которые имеют отношение к анализу, то есть переменные, которые входят в модель для прогнозирования. и удалить все переменные, не относящиеся к прогнозу.

Выбор основных значимых предикторов, которые могут объяснить бедность, является важной задачей, которая по существу зависит от их связи с состоянием бедности в семье и от некоторых знаний, предлагаемых в литературе.

В R выбор предикторов выполняется с использованием команды подмножества `subset`, за которой следует название набора данных и аргумент `select`, который является выражением, указывающим на то, какие столбцы быть выбраны. Далее представлен выбор предикторов на 2016 год. Этот новый набор данных (с названием `data_forecasting_2016.RData`) будет окончательным набором данных, готовым для прогнозирования. Поэтому этот набор данных удобно сохранить с новым названием, чтобы избежать возможных проблем при прогнозировании.

Вставка из R5-5. Выбор предикторов для прогнозирования

```
#####  
# Selection of basic predictors  
#####  
##Selecting the most important predictors among the available  
data_forecasting_2016<-subset(KHS_2016, select=c("hh_code", "weight", "expfact",  
"year", "inc2", "oblast", "Area", "hhage", "hhsex", "Educ4", "Hsize", "NEmpl",  
"NHighEduc", "NProfEduc", "poor", "Rooms", "ElStove", "SatAntenna", "LandLine",  
"Car", "WshMachine", "Ref1", "Ref2", "Freezer", "full.UNEMP", "full.GDP",))  
  
#####  
#Saving the file for forecasting  
#####  
save(data_forecasting_2016, file = "data_forecasting_2016.RData")
```

В следующей таблице представлено описание выбранных предикторов (предсказывающих переменных).

Таблица 5-1 Список переменных в окончательном наборе данных (`data_forecasting_2016.RData`)

Название	Описание	Тип	Источник
hh_code	household ID - Идентификатор домохозяйства	C	Все файлы
weight	вес выборки (уровень населения)	N	KHS_2016.sav
expfact	вес выборки (уровень домохозяйства)	N	KHS_2016.sav
year	год (проведения обследования)	C	KHS_2016.sav
inc2	денежные переводы (тыс.руб.)	N	KHS_2016.sav
oblast	область (места жительства)	C	KHS_2016.sav
Area	район проживания - город (1) или село (2)	B	KHS_2016.sav
hhage	возраст главы домохозяйства	C	KHS_2016.sav

hhhsex	пол главы домохозяйства	B	KIHS_2016.sav
Educ4	уровень образования главы домохозяйства	C	KIHS_2016.sav
Hsize	кол-во членов домохозяйства	C	KIHS_2016.sav
NEmpl	кол-во занятых членов домохозяйства	C	KIHS_2016.sav
NHighEduc	кол-во членов домохозяйствас высшим образованием	C	KIHS_2016.sav
NProfEduc	кол-вочленовдомохозяйстваспрофессиональным образованием	C	KIHS_2016.sav
poor	статус бедности домохозяйства	B	KIHS_2016.sav
Rooms	количество жилых комнат в жилище	C	KIHS_2016.sav
ElStove	наличие электроплиты (стационарной)	B	KIHS_2016.sav
SatAntenna	наличиевдомохозяйствеспутниковой антенны	B	KIHS_2016.sav
LandLine	наличиетелефона (наземной линии)	B	KIHS_2016.sav
Car	наличие автомашины	B	KIHS_2016.sav
WshMachine	наличиестиральной машины	B	KIHS_2016.sav
Ref1	наличие однокамерного холодильника	B	KIHS_2016.sav
Ref2	наличие 2/3 - камерного холодильника	B	KIHS_2016.sav
Freezer	наличие морозильника	B	KIHS_2016.sav
full.UNEMP	уровень безработицы в области проживания	N	Файлы МЭ
full.GDP	валовой региональный продукт на душу населения области	N	Файлы МЭ

Примечание: C=категориальные / дискретные; B= бинарные;
N=числовые непрерывные.

Этот набор данных содержит потенциальные предикторы на уровне домохозяйства и области, бинарную переменную результата (poor) и веса выборки, связанные с каждым домохозяйством в обследовании. Как будет ясно из следующего раздела, предикторы могут быть разных статистических типов: двоичные, категориальные, непрерывные. Эта классификация полезна, когда эти предикторы включены в модель.

Обратите внимание, что набор данных имеет два разных типа весов:

- **expfact** - весовые коэффициенты, которые будут использоваться для экстраполяции из выборки домашних хозяйств на общее количество домашних хозяйств(сумма весов даёт общее количество домохозяйств в стране);
- **weight** - это вес или веса, который следует использовать для экстраполяции из выборки домохозяйств на общую численность населения(сумма весов даёт численность населения в стране)..

Бинарная переменная результата **poor** - бедный равна единице, когда значение регионально дефлированного потребления домохозяйства на душу населения (**pccd**) ниже черты бедности (**pline**) и равна нулю в противном случае.

Переменная **Educ4** представляет уровень образования главы домохозяйства. Эта переменная была получена как совокупность некоторых категорий в первоначальном вопросе обследования. **Educ4** кодируется как **HigherEduc** «Высшее образование», если глава домохозяйства имеет более высокий уровень образования, «Проф. Образование» **Prof.** **Educ** если главад/х имеет неполное высшее или среднее профессиональное

образование или начальное профессиональное техническое образование (с общим средним образованием или без него), и «Среднее или ниже» *Secondary or less* во всех остальных случаях. Другие альтернативные агрегаты (собранные значения) можно было выполнить с помощью команды *ifelse*. Похоже, что это лучшая перекодировка, основанная на распределении исходной переменной и системы образования в стране.

Переменная *inc2* - это сумма (в тысячах сомов) денежных переводов из-за рубежа, полученных домохозяйством. Переменная *Hsize* имеет 7 категорий: категория от 1 до 6 представляет количество членов в домохозяйстве. Категория 7 означает, что в домохозяйстве 7 или более членов. Переменная *Rooms* представляет количество жилых комнат в единице жилья. *Rooms = 6* означает, что в жилой единице шесть или более комнат.

Предиктор *NEmpl* представляет количество занятых членов в домашнем хозяйстве, принимая значения от 0 до 4, где 4 указывает не менее 4 занятых в домашнем хозяйстве. Предикторы *NHighEduc* и *NProfEduc* указывают на количество членов в домохозяйстве с высшим или профессиональным образованием, соответственно.

Все остальные предикторы на уровне домохозяйства в списке являются двоичными, принимая значение, равное 1, если домохозяйство владеет предметом, и ноль, если нет.

Макроэкономические переменные в наборе данных называются *full.UNEMP* и *full.GDP*, и значения одинаковы для всех домохозяйств, проживающих в одном регионе (регион проживания представлен в *oblast*).

Если пользователь не хочет изменять перекодированные переменные и выбор предсказывающих переменных, окончательный набор данных можно получить, просто запустив в консоли RStudio следующую строку:

```
source('C:/YOUR_DIRECTORY/data_2016_MoE.R')
```

обращая внимание на изменение каталога, в котором находятся данные пользователей.

6 Проверка данных

Проверка единицы анализа и кодирования каждой переменной важна для ознакомления с используемым набором данных. R предоставляет несколько полезных команд для описания вашего набора данных:

`str(name_of_dataset)` - компактно отображает внутреннюю структуру набора данных, уже сохраненного в R;

`summary (name_of_dataset)`(макс, мин, медиана, среднее,...) для каждой переменной в наборе данных.

Чтобы лучше проанализировать распределение каждой переменной и потенциальную связь с бедностью, необходимо понять их статистический тип.

6.1 Статистический тип переменных

Предиктор - это предсказывающая переменная, которая объясняет состояние бедности *poverty status*, и которая также называется переменной отклика (зависимой переменной) *response variable* или переменной результата (результатирующая переменная) *outcome variable*. Предикторы могут быть классифицированы в соответствии с их статистическим типом, чтобы иметь возможность правильно видеть их распределение и правильно обрабатывать их в модели прогнозирования.

Переменные могут быть:

- Качественными. Они содержат конечное число категорий или отдельных групп. Они не числовые. Они могут быть далее классифицированы в:
 - Двоичные/ бинарные (или дихотомические) переменные. Бинарные переменные - это переменные только с двумя категориями, то есть, которые принимают только два значения. Например, Мужчина или Женщина, Правда или Ложь и Да или Нет.
 - Категориальные (или полиномиальные) переменные. Категориальная переменная - это переменная с более чем двумя категориями, то есть может иметь более двух значений. Категориальные переменные могут быть упорядоченными или неупорядоченными. Упорядоченные категориальные переменные имеют некоторый порядок, например образование (без образования, начальное, среднее, высшее образование). Неупорядоченные или номинальные категориальные переменные не имеют подразумеваемого порядка, все порядки одинаково значимы. Например, религия (мусульманская, христианская, индуистская, атеист, ...) - это номинальная переменная.
- Числовыми. Они представляют измеримую величину. Они могут быть далее классифицированы в:
 - непрерывные переменные. Непрерывные переменные - это числовые переменные, которые могут содержать любое значение в некотором диапазоне. Например, денежные переменные, такие как доход, являются непрерывными.
 - дискретные переменные. Дискретные переменные - это числовые переменные, которые имеют счетное количество значений (единиц) между любыми двумя значениями. Например, возраст - измеряется годами.

6.2 Распределение численности

Распределение частоты (численности) - это совокупность отдельных значений переменной и количеств событий, то есть сколько раз они происходят, иными словами распределение частоты показывает, как количество распределяется по значениям.

В R основными командами являются таблица `table` и гистограмма `hist`, соответственно. Практическое правило состоит в том, чтобы использовать среднее значение `mean` с числовыми переменными (например, денежными переводами) и / или условными средними (например, денежными переводами по состоянию бедности) и таблицей `prop.table` (таблица относительных частот) для двоичных и категориальных переменных.

Эти команды, однако, не учитывают какую-либо возможную связь между переменными. Эти ассоциации дают представление о том, как выбрать предикторы для модели прогнозирования.

Использование команд `table` и `prop.table` приведено в качестве примера расчета частоты / количества бедных домохозяйств. Примером использования среднего и условного среднего значения является расчет среднего значения денежных переводов и условного среднего значения денежных переводов по уровню бедности.

Вставка из R6-1. Распределение численности

```
table(data_forecasting_2016[, "poor"])
##
##      0      1
## 4032  857

prop.table(table(data_forecasting_2016[, "poor"]))
##
##           0           1
## 0.8247085 0.1752915

mean(data_forecasting_2016[, "inc2"])
## [1] 13335.58

tapply(data_forecasting_2016[, "inc2"], data_forecasting_2016[, "poor"], mean)
##           0           1
## 13810.02 11103.45
```

6.3 Отношения между бедными домохозяйствами и другими переменными

Полезным инструментом для изучения взаимосвязи между статусом бедности домохозяйства, который является бинарной переменной, и категориальными переменными (например, регион проживания, уровень образования, наличие товаров длительного пользования и т. д.) является перекрестная таблица (кросс-табулирование).

В кросс-табуляции категории одной переменной определяют строки таблицы, а категории другой переменной - столбцы. Ячейки таблицы показывают, сколько раз встречается определенная комбинация категорий.

Количество бедных домохозяйств по областям показано в качестве примера. Перекрестная таблица рассчитывается в R с использованием следующего кода:

Вставка из R6-2. Перекрестная таблица

```
table(data_forecasting_2016[, "oblast"], data_forecasting_2016[, "poor"])
##
##           0    1
## 41702 Issykul  558 88
## 41703 Jalal-Abad 504 149
## 41704 Naryn    370 141
## 41705 Batken   382 122
## 41706 Osh      428 97
## 41707 Talas    465 61
## 41708 Chui     520 115
## 41711 Bishkek  590 35
## 41721 Osh city 215 49
```

Приведенная выше таблица показывает, что, например, в Баткенской области 122 домохозяйства в выборке (что соответствует 27,6% домохозяйств в регионе) являются бедными, а остальные 382 (что равно 72,4%) не являются бедными. В городе Бишкек только 35 домохозяйств в выборке являются бедными (5,6%).

6.4 Использование весов выборки

Чтобы экстраполировать оценки на все домохозяйства или на общую численность населения, необходимо принять во внимание веса выборки обследования.

Чистый набор данных имеет два разных типа весов:

- `expfact` - весовые коэффициенты, которые будут использоваться для экстраполяции из выборки домашних хозяйств на общее количество домашних хозяйств;
- `weight` - вес, который следует использовать для экстраполяции из выборки домохозяйств на общую численность населения.

Использование этих весов в анализе имеет решающее значение для точной отчетности статистики. Использование весов в R осуществляется с помощью команды `weighted.mean`.

Например, оценка доли населения в бедности для каждой области получена с использованием следующих кодов:

Вставка из R6-1. Средневзвешенные значения

```
weighted.mean(x=data_forecasting_2016[, "poor"],
w=data_forecasting_2016[, "expfact"])
## [1] 0.1785905

weighted.mean(x=data_forecasting_2016[, "poor"],
w=data_forecasting_2016[, "weight"])
## [1] 0.2552601
```

Вышеприведенные результаты показывают, что предполагаемый процент бедных домохозяйств в стране составляет 17,9%, в то время как уровень бедности в стране, то есть число бедных людей среди всего населения, составляет 25,5%.

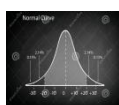
Для оценки доли населения, живущего в бедности в каждой области, используется следующий код:

Вставка из R6-4. Средневзвешенные значения по областям

```
sapply(split(data_forecasting_2016, data_forecasting_2016[, "oblast"]),
function(data_forecasting_2016)
weighted.mean(x=data_forecasting_2016[, "poor"],
w=data_forecasting_2016[, "weight"]))
```

```
##      41702 Issykul 41703 Jalal-Abad      41704 Naryn      41705 Batken
##      0.25662167    0.32368099    0.38082924    0.36822212
##      41706 Osh    41707 Talas    41708 Chui    41711 Bishkek
##      0.22063274    0.18111638    0.30629928    0.09736713
##      41721 Osh city
##      0.24599440
```

Вышеприведенный результат R показывает, что в Нарынской области - самый высокий уровень бедности (38,1% населения, проживающего в Нарыне, является бедным), а в городе Бишкек - самый низкий (9,7%).



Не забудьте использовать соответствующие веса выборки:

Хотите ли вы экстраполировать выборочные оценки на все домохозяйства или на все население?

7 Оценка модели

7.1 Включение в модель предсказывающих переменных

Предикторы $X_1, \dots, X_k, \dots, X_K$ - это предсказывающие переменные, которые объясняют переменную ответа, то есть статус бедности. Выбор основных значимых предикторов, которые могут объяснить бедность, является важной задачей, которая требует анализа связи каждого предиктора с бедностью, как мы видели в предыдущей главе.

Предикторы вводят в модель регрессии другим способом, в соответствии с типом переменной, упомянутым в Разделе 6.1:

- Двоичная / Бинарная переменная вводится в модель как есть. Одна из двух категорий известна как исходная (baseline) или референс (reference) категория, то есть категория сравнения. По умолчанию R выбирает первую категорию. Например, в наборе данных `data_forecasting_2016` переменная «пол главы домохозяйства» (`hhhsex`) имеет две категории: «1 - мужчина» и «2 - женщина» (1 Male 2 Female). Контрольной группой является «1 мужчина». В модели для этой переменной оценивается только один коэффициент, представляющий результат «женщина» по сравнению с «мужчиной».
- Категориальная переменная с k категориями входит в модель как факторная переменная. Например, в наборе данных `data_forecasting_2016` переменная «образование» (`Educ4`) имеет $k = 3$ категории: `Secondaryorless`, `HigherEduc` и `Prof.Educ` («Среднее или меньшее», «Высшее образование» и «Профессиональное образование»). В R эта переменная вводится в модель как `factor(Educ4)`. По умолчанию исходная или референс - категория будет `Secondaryorless` «Среднее или меньшее». В модели по этой переменной оцениваются только два коэффициента, представляющих результат - высшее образование по отношению к наличию среднего или меньшего образования и профессионального образования женщины в отношении получения среднего или меньшего образования.
- Непрерывная числовая переменная (ковариата). Интерпретация коэффициентов регрессии чувствительна к масштабам входных данных. Обычный трюк в регрессии - это «стандартизировать» каждую входную переменную путем вычитания её среднего значения и деления на её стандартное отклонение. Вычитание среднего значения обычно улучшает интерпретацию основных результатов при наличии взаимодействий, а деление на стандартное отклонение ставит все предикторы на одну общую шкалу. Следовательно, непрерывные входные данные должны включаться в модель после стандартизации, особенно когда необходимо понять важность (размер) каждого предиктора в объяснении результата (статуса бедности). Команда в R для стандартизации переменной - это `scale`. Когда, как и в этом анализе, в модели присутствуют бинарные переменные, в недавней литературе (Gelman, 2008) предлагается стандартизировать (перемасштабировать) числовую переменную, посредством вычитания её среднего значения и деления на два стандартных отклонения, что позволяет интерпретировать коэффициенты так же, как и с двоичными входами. Команда в R - `rescale` внутри библиотеки `libraryarm`. Оценочный коэффициент в регрессионной модели будет только один, представляющий степень изменения выходной переменной для каждой 1-й единицы изменения (1-стандартное отклонение, если предсказывающая переменная масштабирована, или 2 стандартных отклонения, если предсказывающая переменная перемасштабирована) в предикторную переменную.

- Дискретная числовая переменная. Перед включением дискретной переменной в регрессионную модель важно решить, следует ли рассматривать её как непрерывный предиктор (ковариат) или категориальный предиктор (фактор). Если дискретная переменная имеет много уровней, то лучше всего рассматривать её как непрерывную переменную (например, «возраст»). Расчетный коэффициент будет только один. Если дискретная переменная принимает всего несколько значений (скажем, менее 10), предиктор можно рассматривать как категориальную переменную и вводить в модель с командой `factor`. В этом случае каждое значение переменной рассматривается как категория, и число оценочных коэффициентов будет равно количеству значений минус 1. Каждый оцененный коэффициент показывает текущее значение по отношению к исходному значению.

Список выбранных предикторов для модели прогнозирования представлен в Таблице 7-1.

Таблица 7-1 Список выбранных предсказывающих переменных, включенных в модель

Название	Описание	Тип	Базовое значение
<i>Демография</i>			
hhhage	Возраст главы д/х	N	-
hhhsex	Пол главы д/х	B	1 – Male (Мужчина)
Hsize	Кол-во членов д/х	C	Hsize=1 (Размер д/х)
<i>Социально-экономические характеристики</i>			
Area	Urban or Rural (город - село)	B	Rural
NHighEduc/Hsize	Доля членов д/х с высшим образованием	N	-
NProfEduc/Hsize	Доля членов д/х с профессиональным образованием	N	-
NEmpl/Hsize	Доля занятых членов в д/х	N	-
<i>Экзогенный (внешний) доход</i>			
inc2/Hsize	Денежные переводы на душу населения (тыс. сом)	N	-
<i>Доступ к услугам, товарам длительного пользования, жилью</i>			
Rooms	Количество жилых комнат в жилье	C	Комнат =1
SatAntenna	Наличие спутниковой антенны	B	0= Нет в наличии
LandLine	Наличие стационарного телефона	B	0= Нет в наличии
Car	Наличие автошины	B	0= Нет в наличии
ElStove	Наличие электроплиты (стационарной)	B	0= Нет в наличии
WshMachine	Наличие стиральной машины - автомата	B	0= Нет в наличии
Ref1	Наличие 1-камерного холодильника	B	0= Нет в наличии
Ref2	Наличие 2-3-х камерного холодильника	B	0= Нет в наличии
<i>Макроэкономические переменные (областной уровень)</i>			
full.GDP	ВВП на душу населения (тысяч)	N	-
full.UNEMP	Уровень безработицы	N	-

Примечание: C= категориальные, B= бинарные, N= числовые непрерывные

Выбранные предсказывающие переменные на уровне домохозяйств сгруппированы в пять отдельных блоков: четыре блока, связанных с предикторами на уровне домохозяйств, и один блок связанный с переменными областного уровня.

Первый блок предсказывающих переменных включает в себя те демографические характеристики, которые могут оказать сильное влияние на риск бедности, особенно количество компонентов домохозяйства. Возраст главы домохозяйства связан со стадией жизненного цикла домохозяйства.

Второй блок отражает характеристики домохозяйства, связанные с рынком труда. Предполагается, что уровень образования, достигнутый главой домохозяйства, должен охватывать атрибуты человеческого капитала домохозяйства. Отношение числа

занятых членов семьи к общему количеству компонентов позволяет измерить нагрузку, которая ложится на занятых в профессии членов домохозяйства.

Третий блок переменных представляет переменные дохода. В качестве предсказывающих переменных используются источники дохода, которые являются экзогенными переменными, то есть независимыми переменными, которые влияют на модель, и при этом модель не влияет на них. Количество денежных переводов и трансфертов социальной помощи (на душу населения) обычно можно классифицировать как внешние переменные и включить в модель. Более того, их можно прогнозировать, используя лучшую доступную информацию «за пределами» модели. Однако только денежные переводы на душу населения в модели являются статистически значимыми.²

Четвёртый блок представляет собой набор контрольных переменных, связанных с общим уровнем жизни домохозяйства. Владение товарами длительного пользования также касается способа измерения потребления. Потребление товаров длительного пользования оценивается за счёт услуг, которые семья получает от товаров, находящихся в её распоряжении в течение соответствующего периода.

Последний блок относится к переменным, которые определяют среду, в которой живут домохозяйства. В частности, на областном (региональном) уровне этими показателями являются уровень безработицы и валовой региональный продукт на душу населения, которые являются экономическими показателями, которые в литературе тесно связаны с уровнем бедности.



Чтобы узнать больше о масштабировании предикторов:

Гельман А. (2008 г.). «Масштабирование регрессионных входных данных путём деления на два стандартных отклонения» (Scaling regression inputs by dividing by two standard deviations), *Statistics in Medicine*, Том 27, с. 2865-2873.

7.2 Оценка модели

Формально, оцениваемая модель - это модель, описанная в уравнении (1), которая воспроизведена здесь для удобства:

$$p_i = \Pr(y_i = 1 | \mathbf{x}) = \text{logit}^{-1}(\beta_0 + \beta_1 x_{1i} + \dots + \beta_k x_{ki} + \dots + \beta_K x_{Ki}) \quad (1)$$

Оценка по существу означает, что необходимо оценить коэффициенты, связанные с предикторами, и, следовательно, можно оценить ожидаемую вероятность бедности для каждого домохозяйства в наборе данных. Оценка параметров получается по максимальной вероятности (см. Текстовая вставка 7-1).

Текстовая вставка 7-1. Оценка по методу максимального правдоподобия

²Выбранный список предикторов не включает общий доход (ни на уровне домохозяйств, ни в расчёте на душу населения). Это связано, по существу, с двумя причинами: во-первых, проблема эндогенности дохода может привести к смещению оценочных параметров модели. В эконометрике эндогенность в широком смысле относится к ситуациям, в которых объясняющая переменная соотносится / коррелируется с величиной погрешности. Эндогенность дохода обусловлена несколькими причинами, одна из которых, ошибки в переменных, была теоретизирована Милтоном Фридманом (1957 г.) в его гипотезе о постоянном или перманентном доходе (PIH). Вторая причина заключается в том, что, когда мы прогнозируем риск бедности, нам необходимо знать, какие предикторы «фиксированы» на горизонте прогноза, а какие должны измениться. Наверняка общий доход семьи может меняться с годами. Поскольку он является предиктором, его необходимо прогнозировать, чтобы прогнозировать риск бедности, то есть нам нужна другая модель для прогнозирования доходов! (Или, если смотреть более реалистично, оценивать необходимо систему одновременных уравнений).

Оценка по методу максимального правдоподобия

В отличие от модели линейной регрессии, которая использует OLS (обычный наименьший квадрат) для оценки параметров, в логистической регрессии используется MLE (оценка максимального правдоподобия). Оценка максимального правдоподобия - это тот набор коэффициентов регрессии, для которого вероятность получения наблюдаемых данных максимальна. Для определения значения параметров берётся логарифм функции правдоподобия, так как он не меняет свойства функции. Логарифмическая правдоподобие дифференцируется, и с помощью итерационных методов, таких как метод Ньютона, определяются значения параметров, которые максимизируют логарифмическую вероятность.



Для тех, кто заинтересован в чтении дополнительной информации об оценке максимального правдоподобия в логистических регрессиях, рекомендуем: Агresti А. (2013 г.), Категориальный анализ данных, третье издание, Wiley. (Agresti A. (2013), Categorical Data Analysis)

Чтобы оценить модель логистической регрессии, нужна правильная функция в R. Основным инструментом, используемым для оценки логистической модели, является функция `glm()`.

```
glm(y ~ x1+x2 + .. +xk + .. +xK,  
family=binomial(link = "logit"), data=the data set)
```

Первый аргумент функции `glm()` - это объект формулы, с заданным двоичным результатом y , за которым следует оператор \sim , а затем предикторы $(x_1 + \dots + x_k + \dots + x_K)$.

Независимая переменная (аргумент) «семья»- `family` добавляется для описания распределения ошибок и функции связи `link`, которая будет использоваться в модели. Обратите внимание, что функция `link` - это команда «logit», которая указывает R выполнить логистическую регрессию.

В соответствии с тем, что было сказано в Разделе 7.1, непрерывные предсказывающие переменные перед вводом в модель стандартизируются на два стандартных отклонения, категориальные предсказывающие переменные включаются с различными уровнями (с использованием команды фактор - `factor`), а двоичные предсказывающие переменные вводятся в модель как есть.

В R, используя команду `glm`, модель можно оценить следующим образом:

Вставка из R7-1. Оценка модели

```
#####  
## Requiring library arm  
#####  
options(repos =c(CRAN = "http://cran.rstudio.com"))  
ipak <-function(pkg){  
  new.pkg <- pkg[!(pkg %in% installed.packages()[, "Package"])]  
  if (length(new.pkg))  
    install.packages(new.pkg, dependencies =TRUE)  
  sapply(pkg, require, character.only =TRUE)  
}  
packages <-c("arm")  
ipak(packages)
```



```
#####
## Setting the working directory and loading the data with the predictors
#####
setwd("C:/YOUR DIRECTORY")
load("data_forecasting_2016.RData")
dim(data_forecasting_2016)

## [1] 4889    26

#####
## Model Estimation
#####
lrfit<-glm(poor~rescale(hhhage) +hhhsex+rescale(I(NEmpl/Hsize))
+rescale(I(NHighEduc/Hsize)) +rescale(I(NProfEduc/Hsize)) + rescale(I(inc2/Hsize))
+factor(Rooms) +factor(Hsize) +rescale(full.GDP) +rescale(full.UNEMP) +Area
+SatAntenna+LandLine +Car+ElStove +WshMachine +Ref1 +Ref2+
Area *rescale(full.UNEMP),
family=binomial(link ="logit"), data=data_forecasting_2016)
```

Текстовая вставка7-2. Библиотекаlibraryarm

Библиотека `library arm`, которая может потребоваться для R, как описано в Разделе 5.1, необходима для функции пере-масштабирования `rescale` и для функции отображения `display`. Функция `rescale` позволяет стандартизировать числовые предикторы по отношению к двукратному стандартному отклонению, в то время как функция `display` суммирует результаты оценочной модели, что позволяет оценивать роль каждого предиктора.

Текстовая вставка7-1. КомандаI()

В R для преобразования переменной перед включением её в модель должна использоваться команда `I()`. Например, чтобы ввести в модель долю занятых членов в домашнем хозяйстве ($NEmpl / Hsize$), переменная (`NEmpl`) должна быть разделена на размер домашнего хозяйства (`Hsize`). В R это делается с помощью `I(NEmpl / Hsize)`.

Текстовая вставка7-4. Член, характеризующий взаимодействие

Влияние каждой предсказывающей переменной на вероятность быть бедным называется *главным эффектом (main effect)*.

Комбинации предсказывающих переменных, в результате, дают член, характеризующий *взаимодействие (interaction term)*. Взаимодействие происходит, если отношение между одной предсказывающей переменной и результатом зависит от значения другой предсказывающей переменной. Взаимодействие вводится в модель путём добавления члена, в котором умножаются два предиктора.

В R взаимодействие представлено символом `*`.

Например, чтобы по-разному понять влияние на уровень бедности уровня безработицы в городских и сельских районах, в модель было введено следующее взаимодействие `Area*rescale(full.UNEMP)`.

7.3 Оценка роли каждой предсказывающей переменной

Исходя из результатов оценочной модели, можно оценить, какие предсказывающие переменные влияют на вероятность бедности положительным

образом, а какие отрицательно влияют на статус бедности. Кроме того, исходя из их размера, также можно понять, какие предикторы являются наиболее важными.

Выходной результат оценки модели R следующий:

Вставка из R7-2. Оценённая модель: коэффициенты и стандартные ошибки

```
#####
display(lrfit,3)
#####

## glm(formula = poor ~ rescale(hhhage) + hhhsex + rescale(I(NEmpl/Hsize)) +
##     rescale(I(NHighEduc/Hsize)) + rescale(I(NProfEduc/Hsize)) +
##     rescale(I(inc2/Hsize)) + factor(Rooms) + factor(Hsize) +
##     rescale(full.GDP) + rescale(full.UNEMP) + Area + SatAntenna +
##     LandLine + Car + ElStove + WshMachine + Ref1 + Ref2 + Area *
##     rescale(full.UNEMP), family = binomial(link = "logit"), data =
data_forecasting_2016)
##
##               coef.est coef.se
## (Intercept)      -5.063    0.754
## rescale(hhhage)   -0.134    0.103
## hhhsex2 Female     0.116    0.111
## rescale(I(NEmpl/Hsize)) -1.283    0.175
## rescale(I(NHighEduc/Hsize)) -1.325    0.196
## rescale(I(NProfEduc/Hsize)) -0.521    0.168
## rescale(I(inc2/Hsize))  -0.451    0.207
## factor(Rooms)2     -0.339    0.254
## factor(Rooms)3     -0.725    0.251
## factor(Rooms)4     -0.769    0.255
## factor(Rooms)5     -0.715    0.279
## factor(Rooms)6     -1.061    0.298
## factor(Hsize)2      2.123    0.749
## factor(Hsize)3      3.172    0.734
## factor(Hsize)4      4.021    0.728
## factor(Hsize)5      4.846    0.727
## factor(Hsize)6      5.535    0.729
## factor(Hsize)7      6.359    0.731
## rescale(full.GDP)   -0.174    0.128
## rescale(full.UNEMP)  0.957    0.138
## AreaUrban          0.167    0.100
## SatAntenna         -0.368    0.141
## LandLine           -0.290    0.157
## Car                -0.585    0.118
## ElStove            -0.776    0.342
## WshMachine         -0.683    0.144
## Ref1               -0.246    0.113
## Ref2               -0.875    0.140
## rescale(full.UNEMP):AreaUrban -0.348    0.188
## ---
##      n = 4889, k = 29
##      residual deviance = 3011.1, null deviance = 4538.7 (difference = 1527.6)
```

Выходные данные показывают по каждому предиктору его оценочный коэффициент (coef.est) и соответствующую оценочную стандартную ошибку (coef.se)

Если знак оценочного коэффициента предиктора положительный, то этот предиктор положительно влияет на вероятность быть бедным в семье. Если знак отрицательный, то этот предиктор отрицательно влияет на вероятность быть бедным.

Стандартная ошибка - это стандартное отклонение оценки. Стандартная ошибка коэффициента показывает, насколько точно модель оценивает неизвестное значение

коэффициента. Стандартная ошибка коэффициента всегда положительна. Используйте стандартную ошибку коэффициента для измерения точности оценки коэффициента. Чем меньше стандартная ошибка, тем точнее оценка.

Коэффициент считается «статистически значимым», если он равен как минимум 2 стандартным ошибкам от нуля по абсолютной величине.

В расчетной модели коэффициент, связанный с владением двухкамерным холодильником, равен

##	coef.est	coef.se
Ref2	-0.875	0.140

Коэффициент статистически значим (так как $|-0.875| > 0.140 \cdot 2 = 0.280$) и отрицателен, что означает, что мы можем быть достаточно уверены, что домохозяйства, имеющие холодильник, имеют меньшую вероятность быть бедными, чем домохозяйства, в которых его нет, при прочих равных условиях.

Используя так называемое «правило $\beta / 4$ » (см. Текстовая вставка 7-5), можно получить представление о том, насколько сильным является влияние каждого предиктора на статус бедности.

Например, давайте рассмотрим коэффициент, связанный с размером домохозяйства, когда размер равен 3:

##	coef.est	coef.se
factor(Hsize)3	3.172	0.734.

Помня о том, что размер домохозяйства, равный единице, является исходным показателем, семья из трех человек имеет максимальную вероятность того, что она будет на 79% ($3.172 / 4$) беднее, чем семья с одним членом, при прочих равных условиях.

Давайте теперь рассмотрим оценочный коэффициент, связанный с региональным уровнем безработицы:

##	coef.est	coef.se
rescale(full.UNEMP)	0.957	0.138

Интерпретация этого коэффициента заключается в том, что два домохозяйства, в остальном совершенно равные по другим характеристикам, но живущие в двух областях, которые отличаются на одну единицу (то есть два стандартных отклонения при пересмотре - см. Раздел 7.1) по уровню безработицы, имеют самую большую разницу в вероятности быть бедным - это 24% ($0.957 / 4$).

Текстовая вставка 7-5. Правило β/4

Как оценить влияние на вероятность быть плохим из-за разницы в 1 единицу в одном из предикторов (например, x_1), удерживая все остальные предикторы фиксированными в некоторых установленных значениях?

Можно вычислить производную логистической кривой по центральному значению предиктора, дифференцируя функцию:

$$\begin{aligned}\frac{\partial}{\partial x_1} \text{logit}^{-1}(\beta_0 + \beta_1 x_1 + \dots + \beta_k x_k + \dots + \beta_K x_K) = \\ = \frac{\partial}{\partial x_1} \text{logit}^{-1}(\mathbf{x}'\beta) = \beta_1 \frac{e^{\mathbf{x}'\beta}}{(1 + e^{\mathbf{x}'\beta})^2}\end{aligned}$$

Например, когда существует только один предиктор x , точка, в которой наклон обратной логистической функции (то есть производная логистической функции) достигает своего максимума при $\alpha + \beta x = 0$, а значение $\text{logit}^{-1}(\alpha + \beta x) = 0.5$ (см. рисунок № ...):

$$\frac{\partial}{\partial x|_{x=0}} = \beta \frac{e^0}{(1 + e^0)^2} = \frac{\beta}{4}.$$

Таким образом $\frac{\beta}{4}$ - это максимальная разница в $\text{Pr}(y_i = 1)$ соответствует разнице в единицах x . Для удобства, мы можем взять коэффициенты логистической регрессии (отличные от постоянного члена) и разделить их на 4, чтобы получить верхнюю границу прогностической разности, соответствующей разнице единиц в x .

Эта верхняя граница является разумным приближением около середины логистической кривой, где вероятности близки к 0,5.

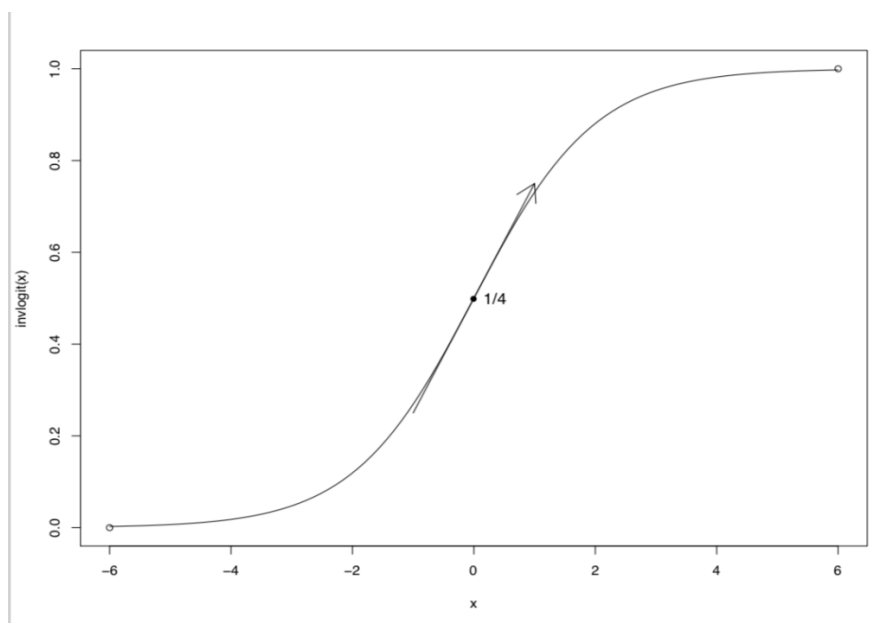


Рисунок 7-1: Функция обратного логита $\text{logit}^{-1}(x)$: преобразование из линейных предикторов в вероятности, которое используется в логистической регрессии.



Чтобы отобразить результаты оценки модели, команда отображения `display` (название модели) - это не единственно возможная команда.

Например, другая популярная команда - сводка `summary`(название модели).

7.4 Оценка точности модели

После того, как логистическая модель была адаптирована к данным, важно проверить, что предполагаемая модель на самом деле является действительной (валидной) моделью. Тщательное изучение степени, в которой подобранная модель обеспечивает надлежащее описание наблюдаемых данных, может быть выполнено с использованием следующих диагностических инструментов:

- Отклонение - *Deviance*;
- Сгруппированные остатки - *Binned residuals*;
- Частота появления ошибок / коэффициент погрешности - *Error rate*.

7.4.1 Отклонение модели

Чтобы оценить, как работает модель логистической регрессии, вместо вычисления суммы квадратов (например, R^2) используется **отклонение - deviance**. Отклонение представляет собой статистическую сводку соответствия модели, определенную для логистической регрессии и других обобщенных линейных моделей.

Отклонение определяется как 2-кратный логарифм функции правдоподобия (с точностью до произвольной аддитивной постоянной).

Для оценки полезны следующие свойства отклонения:

- Отклонение является мерой ошибки; меньшее отклонение означает лучшее соответствие данным.
- Если к модели добавляется неинформативный предиктор (то есть просто случайный шум), мы ожидаем, что отклонение уменьшится на 1.
- Когда в модель добавляется информативный предиктор, мы ожидаем, что отклонение уменьшится более чем на 1. Когда K предикторов добавлено в модель, мы ожидаем, что отклонение уменьшится более чем на K .

Выходной результат оценочной модели, о которой также сообщается ниже, показывает:

```
## ---  
## n = 4889, k = 29  
## residual deviance = 3011.1, null deviance = 4538.7 (difference = 1527.6)
```

«Остаточное отклонение» *residualdeviance* - это отклонение модели, рассчитанное для $n = 4889$ домохозяйств с $K = 29$ предсказывающих переменных.

«Нулевое отклонение» *nulldeviance* - это отклонение нулевой модели, которая представляет собой модель без предикторов и только с постоянным членом β_0 .

Говорят, что модель логистической регрессии с K предикторов (выбранная модель) обеспечивает лучшее соответствие данным, если она демонстрирует улучшение по сравнению с моделью без предикторов (нулевой моделью). Это означает, что есть улучшение, когда значение «остаточного отклонения» модели с K предикторами по меньшей мере на K единиц ниже, чем значение «нулевого отклонения».

В нашем случае, добавив все предикторы ($K = 29$) в модель, отклонение уменьшилось на 1527,6. Это намного больше, чем ожидаемое снижение на 29, если предикторы были бы случайным шумом, поэтому было явное улучшение по отношению к нулевой модели.

7.4.2 Сгруппированные остатки

Остатки для логистической регрессии определяются как наблюдаемые значения минус ожидаемые значения:

$$\text{residual}_i = y_i - E(y_i | \mathbf{x}_i) = y_i - \text{logit}^{-1}(x'_i \beta).$$

Данные y_i дискретны, как и остатки. Например, если $\text{logit}^{-1}(x'_i \beta) = 0.7$, то $\text{residual}_i = -0.7$ или $+0.3$, в зависимости от того, $y_i = 0$ или 1 . В результате графики необработанных остатков от логистической регрессии, как правило, бесполезны. Вместо этого можно построить график *сгруппированных остатков* - *binned residuals*, разделив данные на категории (группы, или бины) на основе их оценочных значений, а затем построить график среднего остатка в зависимости от усредненного значения для каждого элемента.

Команда в R для построения графиков остатков `-binnedplot` в библиотеке `libraryarm`, а фрагмент кода для определения остатков и построения графиков выглядит следующим образом.

Вставка из R7-3. Сгруппированные остатки(`binnedresiduals`)

```
#####  
## Residuals  
#####  
## Estimated values  
pred <- lrfit$fitted.values  
## Observed values  
y <- data_forecasting_2016$poor  
## residuals  
res <- y - pred  
  
#####  
## The binned residuals plot  
#####  
binnedplot(pred, res, nclass=60,  
xlab="Expected Values", ylab="Average residual",  
main="Binned residual plot")
```

График представлен на Рисунке 7-2, где данные были разделены на 60 бинов одинакового размера. При выборе количества бинов существует некоторая произвольность: каждый бин должен содержать достаточно точек, чтобы усредненные остатки были не слишком нестабильными, но это помогает включить много бинов, чтобы увидеть больше трендов в остатках. В нашем примере 60 бинов дали достаточное разрешение, но при этом у них было достаточно точек на бин.

Пунктирные линии (рассчитанные как $2\sqrt{p(1-p)/B}$, где B - это количество точек на бин, $B = \frac{\text{количество наблюдений}}{\text{количество бинов}}$) указывает на границы

стандартной погрешности ± 2 , в пределах которых можно было бы ожидать, что туда попадет около 95% от сгруппированных остатков, если модель действительно верна. Все 60 сгруппированных остатков на Рисунке 7-2 попадают в эти границы.

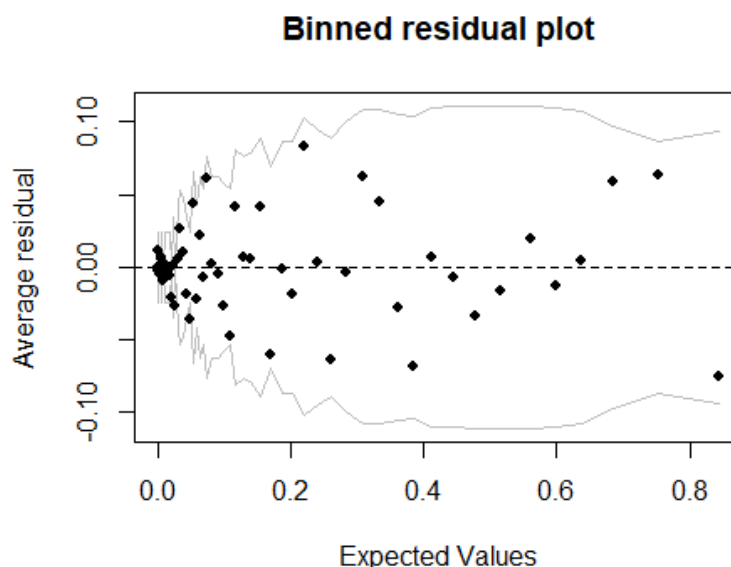


Рисунок7-2. График сгруппированных остатков для модели со взаимодействиями. Группы (бины) расположены неравномерно; скорее, каждыйбин имеет одинаковое количество точек данных. Светлые линии на графике сгруппированных остатков показывают теоретические 95% -ные границы погрешности.

7.4.3 Вероятность ошибки

Таблица классификации (также называемая «матрицей смещения» или «матрицей несоответствий»- *confusionmatrix*) используется для оценки точности прогнозирования модели логистической регрессии.

Когда установлен порог, равный 0,5, все оценочные вероятности, превышающие 0,5, приводят к прогнозируемым значениям, равным 1 (то есть, когда прогнозируемая вероятность бедности превышает 0,5, тогда домохозяйство прогнозируется как бедное). В противном случае, когда предполагаемая вероятность ниже или равна 0,5, тогда прогнозируемое значение будет равно 0 (то есть домохозяйство оценивается не бедное согласно модели).

Таким образом, двоичные наблюдаемые значения результата и двоичные предсказанные значения могут быть перекрестно классифицированы в таблице два на два. Таблица имеет следующую форму.

Таблица7-1: Таблица перекрестной классификации

Наблюдаемое	Прогноз	
	1	0
0	TRUE POSITIVE (a) Истинно положительный (a)	FALSE NEGATIVE (b) Истинно отрицательный (b)
1	FALSE POSITIVE (c) Ложноположительный (c)	TRUE NEGATIVE (d) Ложноотрицательный (d)

Истинно отрицательным (Truenegative) является количество случаев, в которых наблюдаемое значение равно 0 (домохозяйство не бедное), и оценочное значение также равно 0 (домохозяйство прогнозируется как не бедное). Аналогично, Истинно положительным (truepositive) является количество случаев, в которых наблюдаемое значение равно 1 (домохозяйство бедное), и оценочное значение также равно 1 (домохозяйство прогнозируется как бедное). В этих двух ситуациях прогноз модели является правильным.

И наоборот, Ложноположительный результат (falsepositive) - это число случаев, в которых наблюдаемое значение равно 0 (домохозяйство не бедное), но оценочное значение равно 1 (домохозяйство прогнозируется как бедное). Ложноотрицательный (Falsenegative) - это число случаев, в которых наблюдаемое значение равно 1 (домохозяйство бедное), но оценочное значение равно 0 (домохозяйство прогнозируется как не бедное). В последних двух ситуациях прогнозы модели неверны.

Соответственно, коэффициент погрешности *errorrate* определяется как доля (процент) неверных прогнозов (ложноотрицательных + ложноположительных) по отношению к общему числу случаев:

$$error\ rate = \frac{b+c}{a+b+c+d} * 100$$

Коэффициент погрешности всегда должен быть меньше ½ (иначе это не более полезно, чем подбрасывание монеты), но во многих случаях мы ожидаем, что он будет намного ниже.

Это можно сравнить с коэффициентом погрешности нулевой модели, которая просто назначает одинаковую вероятность каждому y_i . Как уже говорилось, нулевая модель имеет только постоянный член, и предполагаемая вероятность будет долей единиц в модели, а именно $p = \sum_{i=1}^n \frac{y_i}{n}$ (напоминаем, что каждый $y_i=0$ или 1). В этом случае коэффициент погрешности нулевой модели будет равен p или $1-p$, в зависимости от того, какой из них меньше. Если коэффициент погрешности модели с K предикторами ниже, чем p , то это показывает улучшение по сравнению с нулевой моделью.

В R коды для матрицы смешения и коэффициента погрешности следующие:

Вставка из R7-4. Коэффициент погрешности (errorrate)

```
#####
## The confusion matrix and the error rate
#####
## The confusion matrix
table(y.oss=y, y.tilde=round(pred))

##      y.tilde
## y.oss    0    1
##      0 3848  184
##      1  483  374

##error rate
mean(pred >0.5&y==0) +mean(pred <0.5&y==1)
## [1] 0.1364287

##error rate of the null model
mean(y)
## [1] 0.1752915
```

Обратите внимание, что Коэффициент погрешности равен

$$errorrate = \frac{b+c}{a+b+c+d} * 100 = \frac{184+483}{3848+184+483+374} * 100 = 13.64\%,$$

что ниже, чем коэффициент ошибок нулевой модели, равный 17,53%. Таким образом, подходящая модель была найдена и готова к прогнозированию.

8 Прогнозирование бедности

8.1 Оценка уровня бедности в пределах выборки

Первым шагом в прогнозировании, который также можно рассматривать как ещё один инструмент проверки / валидации, является так называемое прогнозирование уровня бедности «в пределах выборки».

При прогнозировании распространенной и настоятельно рекомендуемой практикой является обратное преобразование переменных, если они перед этим были стандартизированы, то есть ввод числовых переменных в модель прогнозирования в том виде, как они есть. Стандартизация является полезным инструментом для понимания роли каждого предиктора. Однако на этапе прогнозирования пользователю больше не нужно интерпретировать предикторы, а только использовать их для прогнозирования результата в будущем.

Следовательно, модель будет такой же, но непрерывные переменные больше не стандартизированы. Это будет модель, которая используется для «прогнозирования».

Прежде чем оценивать модель, нам нужно запросить библиотеки для импорта данных (rio) и для отображения оценочных коэффициентов модели (arm).

Вставка из R8-1. Импорт библиотек

```
#####  
## Importing the libraries  
#####  
options(repos =c(CRAN = "http://cran.rstudio.com"))  
  
ipak <-function(pkg){  
  new.pkg <-pkg[!(pkg %in%installed.packages()[, "Package"])]  
  if (length(new.pkg))  
  install.packages(new.pkg, dependencies =TRUE)  
  sapply(pkg, require, character.only =TRUE)  
}  
  
packages <-c("arm", "rio")  
ipak(packages)
```

После установки библиотеки rio с помощью команды import можно загрузить данные.

Вставка из R8-2. Загрузка данных

```
#####  
## Setting the working directory and loading the data with the predictors  
#####  
setwd("C:/YOUR DIRECTORY")  
load("data_forecasting_2016.RData")  
dim(data_forecasting_2016)  
## [1] 4889 26
```

Наконец, модель прогнозирования может быть оценена и отображена.

Вставка из R8-3. Построение модели прогнозирования

```
#####  
## The forecasting model  
#####  
lrfit.forecast<-glm(poor~hhhage +hhhsex +I(NEmpl/Hsize) +I(NHighEduc/Hsize) +  
I(NProfEduc/Hsize) +I(inc2/Hsize) +factor(Rooms) +factor(Hsize)  
+full.GDP +full.UNEMP +Area +  
SatAntenna+LandLine +Car+ElStove +WshMachine +Ref1 +Ref2+  
Area *full.UNEMP,  
family=binomial(link = "logit"), data=data_forecasting_2016)
```

```
display(lrfit.forecast)

## glm(formula = poor ~ hhhage + hhhsex + I(NEmpl/Hsize) + I(NHighEduc/Hsize) +
##      I(NProfEduc/Hsize) + I(inc2/Hsize) + factor(Rooms) + factor(Hsize) +
##      full.GDP + full.UNEMP + Area + SatAntenna + LandLine + Car +
##      ElStove + WshMachine + Ref1 + Ref2 + Area * full.UNEMP, family = binomial(link =
"logit"),
##      data = data_forecasting_2016)
##               coef.est coef.se
## (Intercept)      -4.91    0.82
## hhhage              0.00    0.00
## hhhsex2 Female       0.12    0.11
## I(NEmpl/Hsize)     -1.83    0.25
## I(NHighEduc/Hsize) -2.66    0.39
## I(NProfEduc/Hsize) -1.15    0.37
## I(inc2/Hsize)        0.00    0.00
## factor(Rooms)2      -0.34    0.25
## factor(Rooms)3      -0.73    0.25
## factor(Rooms)4      -0.77    0.25
## factor(Rooms)5      -0.72    0.28
## factor(Rooms)6      -1.06    0.30
## factor(Hsize)2       2.12    0.75
## factor(Hsize)3       3.17    0.73
## factor(Hsize)4       4.02    0.73
## factor(Hsize)5       4.85    0.73
## factor(Hsize)6       5.54    0.73
## factor(Hsize)7       6.36    0.73
## full.GDP             0.00    0.00
## full.UNEMP           0.21    0.03
## AreaUrban            0.74    0.33
## SatAntenna          -0.37    0.14
## LandLine            -0.29    0.16
## Car                 -0.59    0.12
## ElStove             -0.78    0.34
## WshMachine          -0.68    0.14
## Ref1                -0.25    0.11
## Ref2                -0.87    0.14
## full.UNEMP:AreaUrban -0.08    0.04
## ---
##      n = 4889, k = 29
##      residual deviance = 3011.1, null deviance = 4538.7 (difference = 1527.6)
```



Обратите внимание, что модель прогнозирования такая же, как в Разделе 7.3, но непрерывные предикторы `NEmpl/Hsize`, `inc2/Hsize`, `full.UNEMP`, `full.GDP` больше не стандартизированы, и они вводятся в модель в том виде, как они есть. Однако функция `I()` все еще необходима, когда предиктор получен как отношение двух переменных, как в `I(NEmpl/Hsize)`, and `I(inc2/Hsize)`.



Обратите внимание, что характеристики модели прогнозирования (отклонение, сгруппированные остатки, коэффициент погрешности и т. д.) идентичны предыдущим, поэтому нет необходимости в дальнейшей проверке. Это связано с тем, что стандартизация является лишь вопросом интерпретации, но не влияет на подгонку модели. Для получения дополнительной информации по этому вопросу см. Gelman and Hill (2007 г.), Анализ данных с использованием регрессионных и многоуровневых / иерархических моделей, Издательство Cambridge University Press. (Gelman and Hill (2007), Data Analysis Using Regression and Multilevel/Hierarchical Models)

Уровень бедности в стране - *povertyrate*(PR), прогнозируемый моделью в 2016 году (оценка внутри выборки), представляет собой средневзвешенное значение оценочной вероятности быть бедным:

$$\widehat{PR} = \frac{\sum_{i=1}^n \widehat{p}_i \cdot w_i}{\sum_{i=1}^n w_i}$$

где:

\widehat{p}_i - это вероятность домохозяйства i быть бедным, которая оценивается моделью, как указано во Вставке из R 8-4, то есть:

$$\widehat{p}_i = \Pr(y_i = 1|x) = \text{logit}^{-1}(\beta_0 + \beta_1 x_{1i} + \dots + \beta_k x_{ki} + \dots + \beta_K x_{Ki})$$

где $\beta_0, \beta_1, \dots, \beta_k, \dots, \beta_K$ - это оценочные значения бета-коэффициентов;

w_i - это вес выборки (переменная вес - weight в наборе данных), ассоциированный с домохозяйством i .

Модель прогнозирует национальный показатель бедности PR для года выборки (2016 г.) с использованием следующего кода:

Вставка из R8-4. Прогнозирование в пределах выборки (In-sample forecasting)

```
#####
## IN-SAMPLE FORECASTING
## The estimated poverty rate in 2016
#####

EstimatedPoverty<-lrfit.forecast$fitted.values
EstimatedPovertyRate<-weighted.mean(EstimatedPoverty,
data_forecasting_2016$weight)
EstimatedPovertyRate

## [1] 0.2571287
```

что привело к прогнозу уровня бедности, равному 25,7%, что очень близко к официальному уровню бедности 2016 года (25,4%).

Для каждой области расчетные региональные показатели бедности на 2016 год получены как взвешенное среднее значение вероятности бедности этих домохозяйств в выборке, проживающей в данной области. Весы выборки необходимы для репрезентации всего населения. Следующий фрагмент кода в R даёт 9 оценочных региональных уровней бедности.

Вставка из R8-5. Прогнозирование в пределах выборки по областям

```
#####
## Estimation by Oblast
#####

data_Oblast<-data.frame(oblast=data_forecasting_2016$oblast, EstimatedPoverty,
Weights=data_forecasting_2016$weight)
pred_Oblast<-sapply(split(data_Oblast, data_Oblast[, "oblast"]),
function(data_Oblast) weighted.mean(x=data_Oblast[, "EstimatedPoverty"],
w=data_Oblast[, "Weights"]))
pred_Oblast

##      41702 Issykul 41703 Jalal-Abad      41704 Naryn      41705 Batken
##      0.2810749      0.2952394      0.3652402      0.4085268
##      41706 Osh      41707 Talas      41708 Chui      41711 Bishkek
##      0.2603589      0.1764677      0.2376529      0.1288746
##      41721 Osh city
##      0.2407647
```

Результаты могут быть окончательно сопоставлены с наблюдаемыми уровнями бедности на областном уровне, как показано во Вставке из R 8-8. Пример во Вставке из R показывает код для объединения результатов, которые также представлены в Таблице 7-3.

Вставка из R8-6. Сравнение наблюдаемого и предполагаемого уровня бедности в 2016 г.

```
#####
## Combining observed poverty rates with estimated poverty rates by Oblast
## Oblast poverty rates using the raw data
#####
observed_poverty_rates<-sapply(split(data_forecasting_2016,
data_forecasting_2016[, "oblast"]),
function(data_forecasting_2016) weighted.mean(x=data_forecasting_2016[, "poor"],
w=data_forecasting_2016[, "weight"]))
Comparing_poverty_rates<-cbind(EstimatedPovRts=round(pred_Oblast*100, 1),
ObservedPovRts=round(observed_poverty_rates*100, 1))
Comparing_poverty_rates

##              EstimatedPovRts ObservedPovRts
## 41702 Issykul             28.1           25.7
## 41703 Jalal-Abad          29.5           32.4
## 41704 Naryn               36.5           38.1
## 41705 Batken              40.9           36.8
## 41706 Osh                 26.0           22.1
## 41707 Talas               17.6           18.1
## 41708 Chui                23.8           30.6
## 41711 Bishkek             12.9            9.7
## 41721 Osh city            24.1           24.6
```

Таблица8-1.Сравнение наблюдаемого и оценочного уровня бедности в 2016 году

Область	Оценочный уровень бедности	Наблюдаемый уровень бедности
41702 Иссыккуль	28,1	25,7
41703 Джалал-Абад	29,5	32,4
41704 Нарын	36,5	38,1
41705 Баткен	40,9	36,8
41706 Ош	26,0	22,1
41707 Талас	17,6	18,1
41708 Чуй	23,8	30,6
41711 город Бишкек	12,9	9,7
41721 город Ош	24,1	24,6

Простая мера точности расхождений между оценочными и наблюдаемыми значениями - это средняя абсолютная ошибка - *MeanAbsoluteError*(MAE), которая представляет собой среднее абсолютных значений различий между оценочными уровнями бедности (как оценивается в уравнении (№ ...)) и, соответственно, наблюдаемыми (это также официальные показатели бедности на областном уровне, публикуемые НСК). Значение MAE = 2,88, как показано в последней строке ниже, означает, что в среднем разница между наблюдаемым и оценочным уровнем бедности на областном уровне равна 2,88 процентных пункта.

Это можно сделать одной строкой кода, как показано ниже. Несомненно, этот результат приводит к повышению уверенности в хороших результатах модели прогнозирования.

Вставка из R8-7. Средняя абсолютная ошибка (MAE)

```
MAE<-sum(abs(Comparing_poverty_rates[,1]-Comparing_poverty_rates[,2]))/9
MAE
## [1] 2.877778
```

8.2 Прогнозирование уровня бедности вне выборки

В то время как прогнозирование внутри выборки (in-sample forecasting) формально оценивает возможности модели прогнозировать, с использованием данных наблюдений, прогнозирование вне выборки (out-of-sample forecasting) прогнозирует будущие значения с использованием обновлённых значений предикторов. Термин «прогнозирование вне выборки»/ за пределами выборки фактически относится к прогнозированию значений за пределами периода оценки, то есть 2016 года, на основе будущих значений предикторов.

В конечном счёте, будущие значения - это уровни бедности в 2017, 2018 и 2019 годах.

При составлении прогноза важно установить, какие предикторы должны изменяться во времени (time-varying), а какие предикторы являются фиксированными, то есть вряд ли изменятся, по крайней мере, в краткосрочной перспективе.

В этом анализе сделано предположение, что только макроэкономические предикторы (ВВП на душу населения и уровень безработицы) будут меняться со временем. Для этого необходим файл с прогнозом макроэкономических переменных на 2017, 2018 и 2019 годы.

Этот файл с названием `Oblast_dataset.xls` представляет собой файл Excel, который содержит данные о ВВП на душу населения и уровне безработицы в 2015 и 2016 годах на областном уровне, официально опубликованные НСК, а также их прогноз, прогнозируемый Министерством экономики (последнее обновление в августе 2018 года) на 2017, 2018 и 2019 годы. Этот файл был представлен в Текстовой вставке 8.1. То, как прогнозы ВВП на душу населения были получены из прогнозов Министерства экономики по ВВП, показано в Текстовой вставке 8-2.

Как было сказано ранее, необходимо импортировать файл прогнозируемых макропеременных Excel с помощью функции `import` из библиотеки `rio`, что необходимо уже в начале кода:

Вставка из R8-8. Импорт макроэкономического прогноза

```
#####
## Importing the MoE (August 2018) forecasting macro-variables file
#####
macro_var <-import("C:/YOUR DIRECTORY/Oblast_dataset.xls")
```

После того, как прогнозы макропеременных были импортированы в R, необходимо создать новый набор данных для прогнозов.

Новый набор данных содержит как переменные на уровне домохозяйств, которые не должны изменяться, так и прогнозные показатели ВВП на душу населения и уровня безработицы. Эти значения необходимы для каждого года горизонта прогнозирования 2017-2019 гг.

Текстовая вставка8-1. Прогноз ВВП на душу населения

В настоящее время Министерство экономики прогнозирует валовой внутренний продукт (ВВП) в текущих ценах (номинальный сом) и реальный рост ВВП по сравнению с предыдущим годом (индекс) по семи областям, и городам Бишкек и Ош. ВВП на региональном уровне называется Валовой региональный продукт, ВРП. Также приводятся прогнозы численности населения страны.

Как показатель бедности, ВВП по объёму является наиболее подходящим индикатором динамики экономической активности. Поэтому ВРП в постоянных ценах оценивается путём объединения реальных темпов роста. Региональный ВВП на душу населения затем получается путём деления ВРП по объёму на население региона в этом году, предполагая равномерный рост населения по всей стране.

8.2.1 Прогнозирование бедности на 2017 г.

Начиная с 2017 года, будущие значения, согласно прогнозу МЭ, относятся к ВВП на душу населения по объёму и уровню безработицы, в то время как другие предикторы, как предполагается, не изменятся со временем.

Все предикторы (фиксированные и изменяющиеся во времени) должны быть в одной матрице с названием `pred.2017`. В R это можно сделать, обновив в будущем значения двух макроэкономических переменных на 2017 год и оставив микропеременные 2016 года без изменений, следующим образом.

Вставка из R8-9. Обновление матрицы предикторов: 2017 год

```
#####  
## Matrix of predictors for forecasting, year 2017  
#####  
pred.2017<-data.frame(hhhage=data_forecasting_2016$hhhage,  
hhhsex=data_forecasting_2016$hhhsex,  
NEmpl=data_forecasting_2016$NEmpl,  
NHighEduc=data_forecasting_2016$NHighEduc,  
NProfEduc=data_forecasting_2016$NProfEduc,  
inc2=data_forecasting_2016$inc2,  
Rooms=data_forecasting_2016$Rooms,  
Hsize=data_forecasting_2016$Hsize,  
full.GDP=macro_var$GDP_2017[data_forecasting_2016$oblast],  
full.UNEMP=macro_var$Unemp_2017[data_forecasting_2016$oblast],  
Area=data_forecasting_2016$Area,  
SatAntenna = data_forecasting_2016$SatAntenna,  
LandLine=data_forecasting_2016$LandLine,  
Car=data_forecasting_2016$Car,  
ElStove=data_forecasting_2016$ElStove,  
WshMachine=data_forecasting_2016$WshMachine,  
Ref1=data_forecasting_2016$Ref1,  
Ref2=data_forecasting_2016$Ref2)
```

С помощью команды `predict.glm` с параметром `type="response"` можно предсказать для каждого домохозяйства i его вероятность быть бедным в 2017 году, когда присущие ему характеристики не изменятся, зато изменятся экономические переменные области, в которой проживает домашнее хозяйство.



Обратите внимание, что по умолчанию R оценивает вероятность быть бедным в логит-масштабе (то есть прогнозы в терминах *log-odds* - логарифма отношения шансов). Следовательно, `type = "response"` используется для получения прогнозов по шкале вероятностей.

Вставка из R8-10. Прогноз вероятности быть бедным, 2017 год

```
#####  
## Prediction of poverty status in 2017  
#####  
predictions_2017<-predict.glm(lrfit.forecast, pred.2017, type="response")
```

Взвешивая прогнозируемые вероятности быть бедным с весами выборки, связанными с каждым домохозяйством, можно прогнозировать уровень бедности на национальном и областном уровне на 2017 год:

Вставка из R8-11. Прогноз уровня бедности на национальном уровне, 2017 год

```
#####  
## Prediction of national poverty rate in 2017  
#####  
Est.PovertyRate_2017<-weighted.mean(predictions_2017,data_forecasting_2016$weight)  
Est.PovertyRate_2017  
  
## [1] 0.2530801
```

Вставка из R8-12. Прогноз уровня бедности на областном уровне, 2017 год

```
#####  
## Prediction of poverty rates in 2017 at regional (oblast) level  
#####  
  
data_Oblast$predictions_2017=predictions_2017  
pred_Oblast_2017<-sapply(split(data_Oblast, data_Oblast[, "oblast"]),  
function(data_Oblast)  
weighted.mean(x=data_Oblast[, "predictions_2017"], w=data_Oblast[, "Weights"]))  
  
forecast_2017<-as.matrix(round(pred_Oblast_2017*100, 2), 9, 1)  
colnames(forecast_2017)="Estimated values 2017"  
forecast_2017  
  
## Estimated values 2017  
## 41702 Issykul 27.67  
## 41703 Jalal-Abad 29.05  
## 41704 Naryn 36.23  
## 41705 Batken 40.51  
## 41706 Osh 25.66  
## 41707 Talas 17.33  
## 41708 Chui 23.30  
## 41711 Bishkek 12.57  
## 41721 Osh city 23.47
```

Текстовая вставка8-2. Разработки: корректировка весов выборки

Вес выборки, привязанный к каждому домохозяйству, также может измениться. Можно скорректировать изменения характеристик населения с течением времени, создав новый набор весов выборки, чтобы соответствовать набору прогнозируемых контрольных итоговых значений для населения (например, если прогнозируется, что размер домохозяйств изменится, то новые веса позволят учесть влияние этого изменения на состав домохозяйства).

Текстовая вставка 8-3. Другие возможные сценарии

Предложенный инструмент прогнозирования также позволяет лицам, определяющим политику, делать прогнозирование на основе сценариев типа «а что, если». В рамках концепции «что, если» полученные прогнозы не рассматриваются как вероятный результат, а основаны на ситуациях, которые могут произойти. Например, каким был бы уровень бедности, если бы объём денежных переводов из-за рубежа увеличился на определённый процент? Или что будет с уровнем бедности, если начнётся неожиданный экономический рост? Построение прогнозов на основе сценариев позволяет создавать широкий спектр возможных прогнозов и определять некоторые крайние варианты. Например, обычно представляется «лучший», «средний» и «наихудший» сценарии, хотя может быть сгенерировано много других сценариев. Осмысливание и документирование таких противоположных крайних сценариев может привести к раннему планированию на случай непредвиденных обстоятельств (Гиндман и Афанасопулос, 2018 г.). Таким образом, целью анализа сценариев не является попытка показать одну точную картину будущего. Вместо этого представляется несколько альтернативных будущих разработок (иногда называемых «альтернативными мирами»).

8.2.2 Прогнозирование бедности на 2018 г.

Аналогичным образом можно оценить уровень бедности на 2018 год, выполнив те же шаги, что и для 2017 года.

Во-первых, необходим новый набор данных предикторов на 2018 год. Это делается в R, путём обновления будущих значений двух макроэкономических переменных в 2018 году, при этом микропеременные 2016 года остаются неизменными.

Вставка из R8-13. Обновление матрицы предсказывающих переменных: 2018 год

```
#####  
## Matrix of predictors for forecasting, year 2018  
#####  
  
pred.2018<-data.frame(hhhage=data_forecasting_2016$hhhage,  
hhhsex=data_forecasting_2016$hhhsex,  
NEmpl=data_forecasting_2016$NEmpl,  
NHighEduc=data_forecasting_2016$NHighEduc,  
NProfEduc=data_forecasting_2016$NProfEduc,  
inc2=data_forecasting_2016$inc2,  
Rooms=data_forecasting_2016$Rooms,  
Hsize=data_forecasting_2016$Hsize,  
full.GDP=macro_var$GDP_2018[data_forecasting_2016$oblast],  
full.UNEMP=macro_var$Unemp_2018[data_forecasting_2016$oblast],  
Area=data_forecasting_2016$Area,  
SatAntenna = data_forecasting_2016$SatAntenna,  
LandLine=data_forecasting_2016$LandLine,
```



```
Car=data_forecasting_2016$Car,
ElStove=data_forecasting_2016$ElStove,
WshMachine=data_forecasting_2016$WshMachine,
Ref1=data_forecasting_2016$Ref1,
Ref2=data_forecasting_2016$Ref2)
```

Затем с помощью команды `predict.glm` с параметром `type="response"` делаются новые прогнозы состояния бедности для каждого домохозяйства на 2018 год.

Вставка из R8-14. Прогноз вероятности быть бедным, 2018 год

```
#####
## Prediction of poverty status in 2018
#####
predictions_2018<-predict.glm(lrfit.forecast, pred.2018, type ="response")
```

Наконец, взвешивая прогнозируемые вероятности бедности с весами выборки, связанными с каждым домохозяйством, делаются прогнозы уровня бедности на национальном и областном уровне на 2018 год.

Вставка из R8-15. Прогноз уровня бедности на национальном уровне, 2018 год

```
#####
## Prediction of national poverty rate in 2018
#####
Est.PovertyRate_2018<-weighted.mean(predictions_2018,
data_forecasting_2016$weight)
Est.PovertyRate_2018
## [1] 0.2507121
```

Вставка из R8-16. Прогноз уровня бедности на областном уровне, 2018 год

```
#####
## Prediction of poverty rates for 2018 at regional (oblast) level
#####
data_Oblast$predictions_2018=predictions_2018
pred_Oblast_2018<-sapply(split(data_Oblast, data_Oblast[, "oblast"]),
unction(data_Oblast)
weighted.mean(x=data_Oblast[, "predictions_2018"], w=data_Oblast[, "Weights"]))

forecast_2018<-as.matrix(round(pred_Oblast_2018*100, 2), 9, 1)
colnames(forecast_2018)="Estimated values 2018"
forecast_2018

## Estimated values 2018
## 41702 Issykul 27.56
## 41703 Jalal-Abad 28.75
## 41704 Naryn 35.90
## 41705 Batken 40.22
## 41706 Osh 25.37
## 41707 Talas 17.09
## 41708 Chui 23.05
## 41711 Bishkek 12.48
## 41721 Osh city 23.28
```

8.2.3 Прогнозирование бедности на 2019 г.

На 2019 год оценивается только национальный уровень бедности, поскольку неопределённость в оценках будет слишком высокой, чтобы полагаться на оценочные значения.

Набор данных предикторов для 2019 года должен быть построен следующим образом.

Вставка из R8-17. Обновление матрицы предсказывающих переменных: 2019 год

```
#####  
## Matrix of predictors for forecasting, year 2019  
#####  
  
pred.2019<-data.frame(hhhage=data_forecasting_2016$hhhage,  
hhhsex=data_forecasting_2016$hhhsex,  
NEmpl=data_forecasting_2016$NEmpl,  
NHighEduc=data_forecasting_2016$NHighEduc,  
NProfEduc=data_forecasting_2016$NProfEduc,  
inc2=data_forecasting_2016$inc2,  
Rooms=data_forecasting_2016$Rooms,  
Hsize=data_forecasting_2016$Hsize,  
full.GDP=macro_var$GDP_2019[data_forecasting_2016$oblast],  
full.UNEMP=macro_var$Unemp_2019[data_forecasting_2016$oblast],  
Area=data_forecasting_2016$Area,  
SatAntenna = data_forecasting_2016$SatAntenna,  
LandLine=data_forecasting_2016$LandLine,  
Car=data_forecasting_2016$Car,  
ElStove=data_forecasting_2016$ElStove,  
WshMachine=data_forecasting_2016$WshMachine,  
Ref1=data_forecasting_2016$Ref1,  
Ref2=data_forecasting_2016$Ref2)
```

Затем делаются новые прогнозы статуса бедности для каждого домохозяйства на 2019 год.

Вставка из R8-18. Прогноз вероятности быть бедным, 2019 год

```
#####  
## Prediction of poverty status in 2019  
#####  
predictions_2019<-predict.glm(lrfit.forecast, pred.2019, type ="response")
```

Наконец, взвешивая прогнозируемые вероятности быть бедными с весами выборки, связанными с каждым домохозяйством, делаются прогнозы уровня бедности на национальном и областном уровне на 2019 год.

Вставка из R8-19. Прогноз уровня бедности на национальном уровне, 2019 год

```
#####  
## Prediction of national poverty rate for 2019  
#####  
predictions_2019<-predict.glm(lrfit.forecast, pred.2019, type ="response")  
Est.PovertyRate_2019<-weighted.mean(predictions_2019,  
data_forecasting_2016$weight)  
Est.PovertyRate_2019  
## [1] 0.2451889
```



Чтобы экспортировать результаты в виде файла Excel, командой в R является `write.csv ()`. В этом случае может быть удобно собрать все оценочные значения в одном файле с помощью команды `cbind ()`.

Вставка из R8-20. Экспорт результатов в файл Excel

```
#####
Combining results at national level
#####
National_PovRates<-cbind(EstimatedPovRate2016=round(EstimatedPovertyRate*100, 2),
EstimatedPovRate2017=round(Est.PovertyRate_2017*100, 2),
EstimatedPovRate2018=round(Est.PovertyRate_2018*100, 2),
EstimatedPovRate2019=round(Est.PovertyRate_2019*100, 2))
colnames(National_PovRates)<-c("PR_2016", "PR_2017", "PR_2018", "PR_2019")
National_PovRates
##      PR_2016 PR_2017 PR_2018 PR_2019
## [1,]  25.71  25.31  25.07  24.52

#####
Saving the results as an Excel (.csv) file
#####

write.csv(National_PovRates, "National_PovRates.csv")
```

9 Приложения

9.1 Приложение 1: Начало работы в R / RStudio

R - это бесплатная программная среда с открытым исходным кодом для статистических вычислений и создания графики. **RStudio** - это интегрированная среда разработки с открытым исходным кодом для R, которая добавляет множество функций и инструментов производительности для R.

R - это пакет общего назначения, который включает поддержку широкого спектра современных статистических и графических методов (многие из которых были внесены пользователями). Он доступен для большинства платформ UNIX, Windows и MacOS. «Фонд статистических вычислений R» (R Foundation for Statistical Computing) владеет и администрирует авторские права на программное обеспечение R и документацию. R предоставляется в форме исходного кода в соответствии с условиями Открытого лицензионного соглашения (Генеральной общедоступной лицензии GNU) Фонда бесплатного / свободного программного обеспечения (Free Software Foundation).

RStudio облегчает использование R, объединяя R-справку (help) и документацию, предоставляя браузер рабочей области и средство просмотра данных, а также поддерживает подсветку синтаксиса (т.е. выделение синтаксических элементов яркостью или цветом), автозавершение / автодополнение кода и интеллектуальное отступы - структурированное расположение текста. RStudio - это интегрированная среда разработки (IDE) для R. RStudio включает в себя консоль, редактор с выделением синтаксиса, который поддерживает прямое выполнение кода, а также инструменты для построения графика, истории, отладки и управления рабочей областью.

Пользователи Windows перед запуском должны выполнить следующие шаги:

Сначала необходимо установить R:

1. Перейдите на страницу <http://www.r-project.org>
2. Нажмите на ссылку «загрузить R» (downloadR) в середине страницы в разделе «Начало работы» (GettingStarted).
3. Выберите местоположение CRAN (зеркального сайта) и кликнуть на соответствующей ссылке.
4. Нажмите на ссылку «Загрузить R для Windows» (DownloadRforWindows) в верхней части страницы.
5. Нажмите на ссылку «установить R в первый раз (installRforthefirsttime) в верхней части страницы.
6. Нажмите «Загрузить R для Windows» (DownloadRforWindows) и сохраните исполняемый файл (файл с расширением «.exe») где-нибудь на вашем компьютере. Запустите файл .exe и следовать инструкциям по установке.

Теперь, когда программа R установлена, вам необходимо загрузить и установить RStudio.

Для установки RStudio:

1. Перейдите на страницу <http://www.rstudio.com> и нажмите на ссылку «Загрузить RStudio» (DownloadRStudio).
2. Нажмите на ссылку «Загрузить RStudio Desktop» (DownloadRStudioDesktop)
3. Нажмите на версию, рекомендованную для вашей системы, или для последней версии Windows, и сохраните исполняемый файл .exe. Запустите файл .exe и следуйте инструкциям по установке.
4. Нажмите на ссылку для установщика для вашего компьютера и снова следуйте обычным процедурам установки.

После завершения установки RStudio нажмите на значок (графический символ - иконка) «RStudio», чтобы открыть отображаемое окно.

После запуска RStudio появятся следующие панели по умолчанию (см. Рисунок 9-1):

- Консоль (вся левая часть экрана)
- Рабочая область / История (вкладка в правом верхнем углу)
- Файлы / графики / Пакеты / Помощь (вкладка в правом нижнем углу)

После этого необходимо открыть сценарий, в котором написан и сохранен код: наберите команду `file`, а затем откройте файл `openfile` и выполните поиск сценария на компьютере (см. Рисунок 9-2).

Файлы, содержащие скрипт (сценарий), имеют суффикс `*.R`.



Чтобы получить бесплатное руководство на русском языке, перейдите по адресу <https://cran.r-project.org/>, найдите и нажмите «Внесённый вклад» (Contributed) в левой части страницы, затем нажмите «Другие языки» (Other Languages), прокрутите, пока не найдёте русский язык, и вы можете скачать PDF-версию. руководства, написанного Шипуновым и соавторами.

Рисунок 9-1. Rstudio: панель по умолчанию

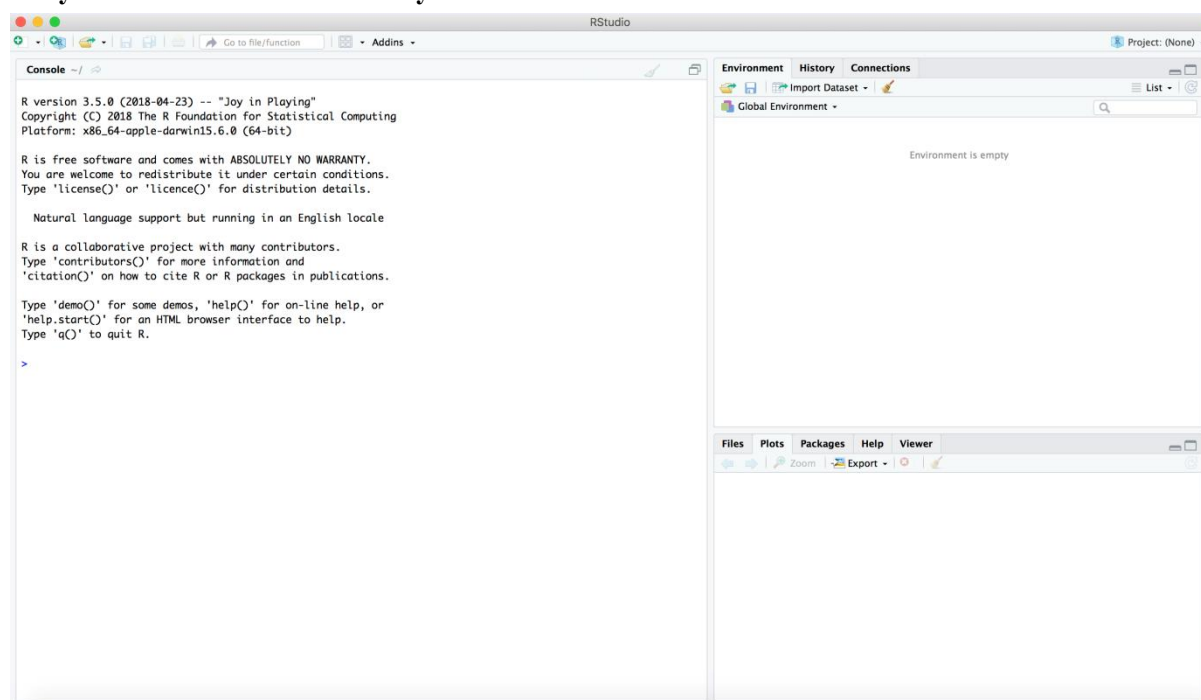
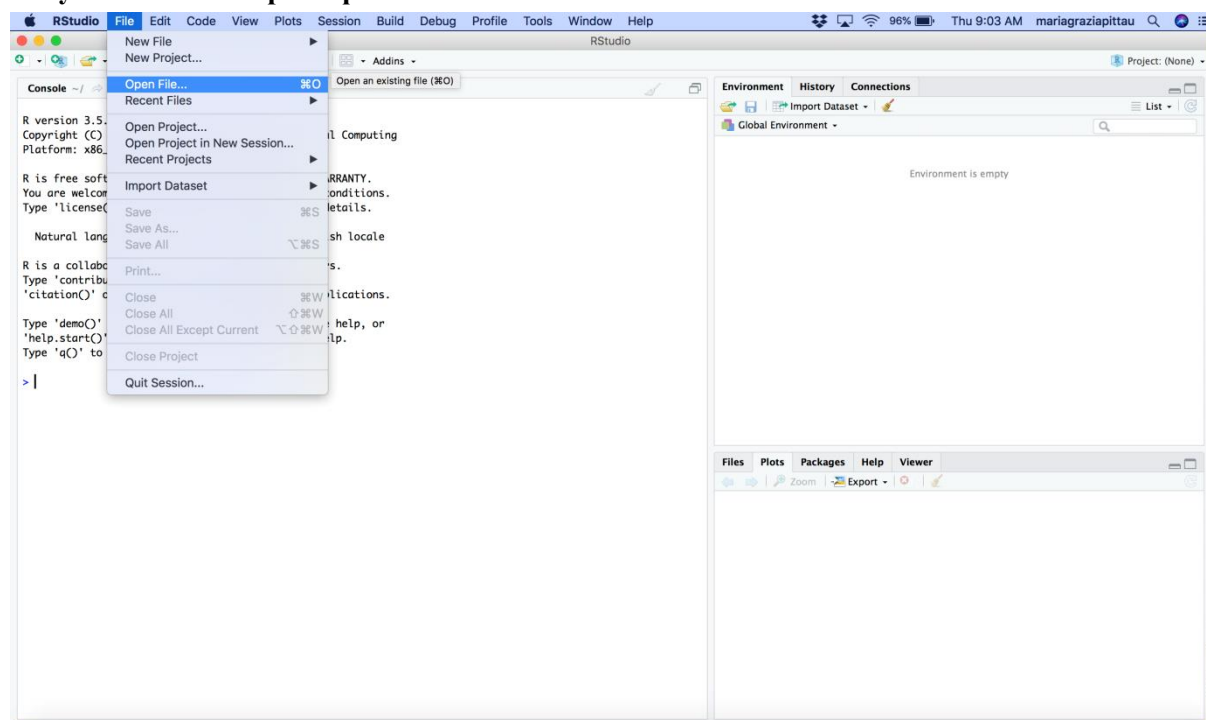
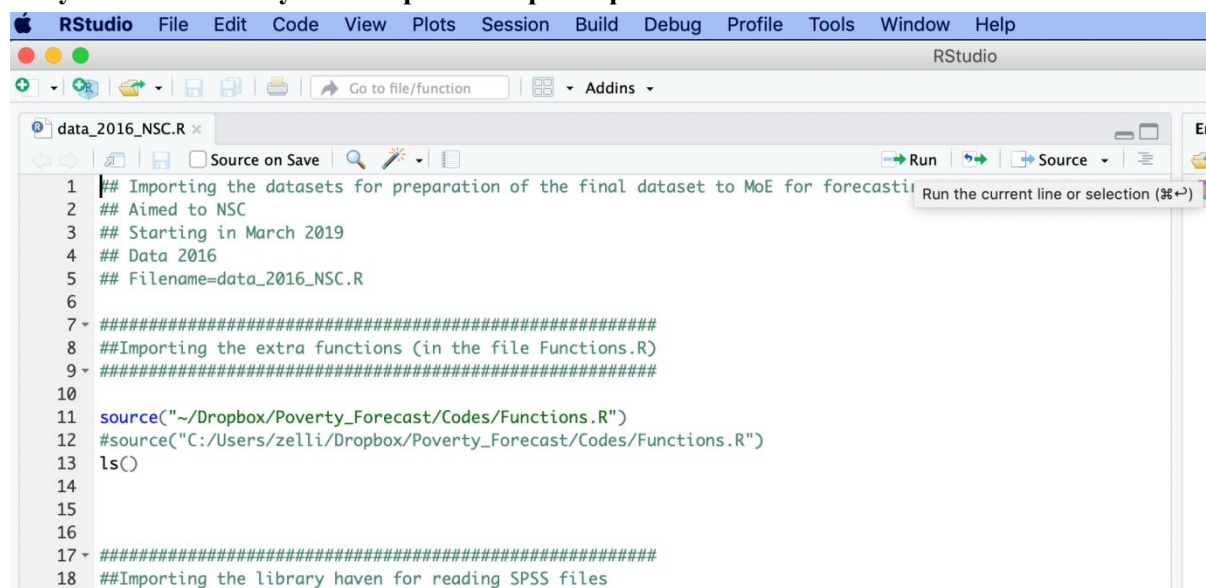


Рисунок9-1.Какоткрытьфайл



После открытия файла сценария для выполнения текущей строки или выделения строк, в которых в данный момент находится курсор, используйте кнопку панели «Выполнить» - **Run** (см. Рисунок 9-3). После выполнения строки кода RStudio автоматически перемещает курсор на следующую строку. Это позволяет пользователю пошагово проходить последовательность строк.

Рисунок9-3. Как запустить строки в скрипт-файле



9.2 Приложение 2: Словарь специальных статистических терминов

- Двоичные / бинарные или дихотомические: Двоичные или Бинарные переменные имеют только два значения. Эти значения обычно кодируются как «0» и «1». Бинарная переменная является исходной переменной логистической регрессии.
- Корреляция: Когда две переменные линейно связаны друг с другом, так что одна систематически изменяется вместе с другой, считается, что они скоррелированы - то есть они взаимосвязаны. Сила и направление корреляции могут быть представлены таким коэффициентом корреляции, как коэффициент Пирсона или Спирмена Ро. Этот коэффициент можно использовать для определения того, насколько сильна связь между переменными и является ли она положительной (когда значение одной и другой переменной увеличивается) или отрицательной (когда значение одной переменной увеличивается, а значение другой - уменьшается).
- Фиктивная переменная (дамми переменная): фиктивные переменные представляют собой серию дихотомических / бинарных переменных, поэтому регрессионный анализ может быть выполнен с использованием категориальной (номинальной или порядковой) переменной с более чем двумя категориями. Одна категория, обычно та, которая содержит наибольшее количество респондентов, обозначена как «справочная» (референс или отсылочная категория) и не имеет фиктивной переменной. Для всех остальных категорий создаётся фиктивная переменная. Единицы кодируются как «1», если они принадлежат конкретной категории каждой фиктивной переменной, и «0», если нет. Единицы, принадлежащие к справочной категории, кодируются как «0» для всех фиктивных переменных.
- Размер воздействия (Effectsize): размер воздействия является стандартизированной мерой силы наблюдаемого воздействия. Он позволяет исследователю измерить величину отношения или различия по стандартизированной шкале.
- Эстиматор (Estimator): метод оценивания или оценочная формула: формула или алгоритм для генерации оценок параметров с учётом соответствующих данных.
- Члены, характеризующие взаимодействие (Interactionterms): попарные произведения «исходных» независимых переменных. Включение членов взаимодействия в регрессию допускает возможность того, что степень влияния x_i на y зависит от значения некоторой другой переменной x_j . Другими словами, x_j модулирует влияние x_i на y . Например, влияние опыта на заработную плату может зависеть от пола работника.
- Отрезок / свободный член (Intercept): Свободный член это прогнозируемое значение исходной переменной в регрессионной модели, когда все объясняющие переменные имеют значение ноль.
- Логистическая регрессия (LogisticRegression): Если упростить, логистическая регрессия - это версия множественной регрессии, где переменная результата является двоичной (дихотомической), то есть возможны только два результата. Модель может использоваться для расчёта вероятности одного из двух результатов, возникающих в данном случае / наблюдении с использованием значений набора известных объясняющих переменных.
- Основной эффект (Maineffect): это эффект, который данная пояснительная переменная оказывает на конечную переменную. В модели основных эффектов нет членов взаимодействия между объясняющими переменными, поэтому основные эффекты представляют собой уникальное влияние каждой объясняющей переменной на результат.
- Мультиколлинеарность (Multicollinearity): ситуация, когда существует высокая степень корреляции между независимыми переменными в регрессионной модели или, в более общем случае, когда некоторые из X s близки к линейным комбинациям других X s. Симптомы включают в себя большие стандартные ошибки и невозможность получения точных оценок параметров. Это не является серьёзной проблемой, если задачей является

прогноз - зато при попытке оценить причинно - следственные связи это является проблемой.

- Нормальное распределение (Normaldistribution): нормальное распределение - это распределение данных, которое группируется вокруг среднего значения. Когда график представлен на гистограмме, он имеет пик и форму «колокола». Известно, что нормально распределённые данные обладают особыми свойствами, которые позволяют нам делать выводы, и поэтому очень полезны для статистического анализа.
- Смещение пропущенной переменной (Omittedvariablebias): смещение в оценке параметров регрессии, которое возникает, когда соответствующая независимая переменная исключается из модели, и эта пропущенная переменная коррелирует с одной или несколькими из включённых переменных.
- Выходная переменная / переменная результата (Outcomevariable). Выходная переменная, которую иногда называют зависимой переменной, - это переменная, которую вы пытаетесь предсказать с помощью регрессионного анализа. Данные одной или нескольких объясняющих переменных используются для прогнозирования результата. Подобно объясняющим переменным, выходные переменные могут быть чем угодно, что исследователь желает исследовать, если это можно выразить количеством и этот результат будет разумным и надёжным.
- Выпадающее значение (Outlier): Выпадающее значение - это точка в данных или случай, которые не соответствуют общему шаблону данных. Выпадающие значения могут смещать статистический анализ и приводить к ошибочным выводам, поэтому важно, чтобы они были идентифицированы и либо удалены, либо изучены более подробно (или и то и другое!).
- Принцип экономии или простоты (Parsimony). Это самый простой и точный способ объяснения. Это принцип, который важен при построении моделей статистической регрессии - мы не хотим усложнять модель путём включения несвязанных или ненужных поясняющих переменных. Мы ищем наименьший набор переменных, которые могут адекватно объяснить наш результат. В этом процессе часто используются статистические критерии, не учитывающие переменные, которые существенно не повышают точность нашего прогноза.
- Качественные данные (QualitativeData): качественные данные менее осязаемы, чем количественные данные, поскольку они охватывают более широкие типы информации. Акцент делается на некотором качестве информации, а не на количестве.
- Количественные данные (Quantitativedata): В общих чертах, количественные данные относятся к числовым данным. Это информация, которая основана на числовых количествах или информации, которая может (и была) определена количественно.
- Коэффициент регрессии (Regressioncoefficient): коэффициент регрессии представляет отношение между данной объясняющей переменной и выходной переменной / переменной результата. В линейных моделях это именно то, насколько изменяется переменная результата после изменения одной единицы в объясняющей переменной. По сути, это градиент линейных отношений. Существует также стандартизированный коэффициент регрессии, который позволяет более точно сравнивать относительные силы взаимосвязи между различными пояснительными переменными и результатом.
- Остатки (Residuals): регрессионный анализ предназначен для того, чтобы позволить исследователю прогнозировать результат на основе данных одной или нескольких объясняющих переменных. Однако модель никогда не бывает абсолютно точной, и каждая фактическая точка данных, вероятно, будет немного отличаться от значения, предсказанного моделью. Остаток - это разница для каждого случая между фактическим результатом и результатом, предсказанным моделью. Часто они называются «ошибками»

































(errors) в прогнозировании, но, возможно, более точно они представляют дисперсию, которая не может быть учтена моделью.

- Выборка (Sample): Выборка - это выбранные элементы или случаи, взятые из совокупности, и используемые для построения выводов об этой совокупности. Выборки могут осуществляться несколькими способами, но они должны быть репрезентативными для популяции, чтобы можно было достичь надёжных и достоверных результатов.
- Уровень значимости (Significance level): для проверки гипотез это наименьшее значение p , для которого мы не будем отклонять нулевую гипотезу. Если мы выберем уровень значимости 1%, мы говорим, что отклоним ноль тогда и только тогда, когда значение p для теста будет меньше 0,01. Уровень значимости также является вероятностью совершения ошибки типа 1 (то есть отклонения истинной нулевой гипотезы).
- Стандартное отклонение (Standard deviation): Стандартное отклонение - это квадратный корень из дисперсии. Как и в случае с дисперсией, оно представляет собой средний разброс (изменчивость) данных, но оно корректируется таким образом, чтобы поддерживать те же единицы измерения, что и исходные данные. По этой причине стандартное отклонение используется чаще, чем дисперсия, как мера изменчивости.
- Стандартная ошибка (Standard error): Стандартной ошибкой является стандартное отклонение статистики распределения выборки. Это может показаться непонятным для непосвящённых, так что давайте скажем иначе. Представьте, что вы хотите узнать среднее значение переменной в данной популяции. Если бы вы делали выборку из популяции несколько раз, вы каждый раз получали бы другое среднее значение, и из этих средних значений вы могли бы получить новое среднее значение и стандартное отклонение. Это стандартное отклонение называется стандартной ошибкой и показывает, насколько точной будет ваша оценка среднего значения по вашей выборке. Низкие значения означают, что вы можете быть достаточно уверены, что значение, полученное из вашей выборки, близко к истинному значению совокупности. Стандартные ошибки могут быть рассчитаны для большого количества статистических данных, а не только для средних значений.
- Единица (Unit): отдельные случаи из популяции известны как единицы. Характер единицы (измерения) зависит от того, что вас интересует, а также от изучаемой вами популяции. Хотя для единицы характерно - что это отдельный человек, они могут быть чем угодно. Например, если вы хотите сравнить разные школы в данном регионе, то популяцией будут все школы из этого региона, а единицами будут конкретные школы, которые попали в выборку.
- Переменная (Variable): Переменная является измеримой характеристикой или атрибутом наблюдений или единиц. Неудивительно, что переменная, как ожидается, будет изменяться в зависимости от случая или времени. Например, пол - это переменная, которая для разных случаев может принимать значения мужской или женский, которые могут кодироваться 0 и 1, соответственно. В регрессионном анализе мы должны различать объясняющие и выходные переменные / переменные результата, причём последние являются явлениями, которые мы хотим объяснить или по которым мы хотим отчитаться.

9.3 Приложение 3: Слияние файлов

НСК предоставляет файлы ИОДХ. Данные поступают в формате *.sav для пользователей SPSS. Как кратко описано в Разделе 3, семь опросников составляют анкету обследования ИОДХ, каждая форма содержит один или несколько разделов (см. Таблицу 3-1). НСК организует первичные файлы данных SPSS в соответствии со структурой вопросника по формам и разделам, как показано ниже.

Рисунок9-4: Структура файлов ИОДХ 2016 г.

Name
 Basic.sav
 f1_nal.sav
 F2_00.sav
 f2_01.sav
 f2_02.sav
 f2_03.sav
 f2_04.sav
 f3_01.sav
 f3_02.sav
 f3_03.sav
 F4.sav
 f6_01.sav
 f6_02.sav
 f6_03.sav
 f6_04.sav
 f6_05.sav
 f6_06.sav
 f6_07.sav
 f6_08.sav
 f6_054.sav
 f6_61.sav
 f6_77.sav
 f6_79.sav
 f6_0211.sav
 f6_713.sav
 f6_716.sav
 f6_7131.sav
 f7_01.sav
 f7_02.sav
 f7_03.sav
 f7_03q2.sav
 PROFIL_2016.sav

Например, f3_01.sav содержит данные из Раздела 1 Формы 3 (Покупка продуктов питания), а f7_02.sav содержит данные, полученные из Раздела 7.2 исходного вопросника. В случае больших форм файлы данных далее подразделяются в соответствии с определенными вопросами.

Кроме этого, НСК предоставляет дополнительный файл PROFIL_2016.sav, который является основным файлом анализа. В этом файле содержится подробная информация о расходах домохозяйства, потреблении, доходе и некоторых социально-

демографических характеристиках домохозяйства и главы домохозяйства, а также черту бедности и показатели бедности.

Переменные стратификации и веса обследования находятся в файле Basic.sav.

Другая соответствующая информация (например, информация об уровне образования всех членов домохозяйства) доступна в оригинальном файле f1_nal.sav.

В рамках этой структуры микроданные, важные для прогнозирования, будут представлены в виде файла .sav, полученного путем объединения на уровне домохозяйств соответствующей информации из:

- PROFIL_2016.sav,
- f1_nal.sav,
- f7_01.sav (жилищные условия),
- f7_02.sav(наличие товаров длительного пользования).

Далее процедура объединения этих файлов в R иллюстрируется шаг за шагом.

9.3.1 Дополнительные функции в R

Для того, чтобы правильно просмотреть данные и правильно объединить файлы, в R были написаны 4 дополнительных функции и сохранены в независимом файле с названием Functions.R, который был запущен до получения данных.

Дополнительные функции:

Функция **sumNA**- которая определяет количество отсутствующих данных для каждой переменной;

Функция **LeUni** - функция, которая помогает проверять количество отдельных домохозяйств в файлах на индивидуальном уровне;

Функция **select**- функция, которая выбирает для каждого домохозяйства товары длительного пользования, принадлежащие домохозяйствам;

Функция **Aggr**- которая сворачивает долговременный файл f7_02.sav в файл, который содержит строки по количеству домохозяйств.

Файл может быть запущен с использованием команды «источник»- **source**. После запуска файла дополнительные функции автоматически загружаются в рабочий каталог пользователя в R. С помощью команды **ls ()** можно проверить, доступны ли все функции.

Вставка из R9-1.Импорт дополнительных функций

```
#importing the extra functions
source("C:/YOUR DIRECTORY/Functions.R")
ls()
## [1] "Aggr" "LeUni" "Select" "sumNA"
```



Файл Functions.R должен быть сохранен в том же каталоге, где хранятся данные.

9.3.2 Импорт файлов SPSS в R: «Библиотечное убежище»(library haven)

Как уже упоминалось, библиотеки R представляют собой наборы функций и данных, разработанных сообществом. Для установки библиотек необходимо подключение к зеркалам CRAN.

Для этого пользователю необходимо запустить следующие строки в R, которые позволяют перейти непосредственно к зеркалам CRAN и, во всей функции, установить библиотекуhaven, предложенную для импорта файлов SPSS.

Вставка из R9-2. Библиотечное убежище (library haven)

```
# connecting to the CRAN mirror
options(repos =c(CRAN ="http://cran.rstudio.com"))
# requiring packages
ipak <-function(pkg){
  new.pkg <-pkg[!(pkg %in%installed.packages()[, "Package"])]
  if (length(new.pkg))
  install.packages(new.pkg, dependencies =TRUE)
  sapply(pkg, require, character.only =TRUE)
} ##end of requiring packages function

packages <-c("haven")
ipak(packages)

## Loading required package: haven

## haven
## TRUE
```



Если библиотека была правильно установлена, пользователь всегда должен получить сообщение TRUE (верно) ниже названия библиотеки.

Если сообщение - FALSE (неверно), значит при загрузке библиотеки возникли проблемы, которые серьезно повлияют на импорт данных SPSS.

9.3.3 Импорт и обработка файла «бедность» - poverty (PROFIL_2016.sav)

Как было объяснено ранее, перед загрузкой данных в R всегда необходимо указать R, где хранятся данные. Это делается с помощью команды `setwd()`. Как только каталог, в котором находятся данные, установлен, можно прочитать набор данных SPSS с помощью команды `read_sav()`.



Преобразовать данные во фрейм данных всегда удобно с помощью команды `data.frame()`. Фрейм данных - это структура, в которой каждый столбец содержит значения одной переменной, а каждая строка содержит один набор значений из каждого столбца.

Данные, хранящиеся во фрейме данных, а не в традиционном матричном формате, могут иметь числовой, факторный или символьный тип по мере появления данных ИОДХ.

После импорта данных настоятельно рекомендуется просмотреть данные (например, первые 4 строки, как указано ниже) и подсчитать для каждой переменной количество пропущенных записей. Это можно сделать с помощью дополнительной функции `sumNA`.

Код для всех этих шагов расписан в следующей Вставке из R.

Вставка из R9-3. Импорт файла «бедность» - poverty

```
## Setting the working directory
setwd("C:/YOUR DIRECTORY")

## FIRST DATASET
## Reading the file in SPSS---as given from National Statistical Office
KIHS_2016_pov<-as.data.frame(read_sav("PROFIL_2016.sav"))
KIHS_2016_pov[1:4,]

## Looking for missing data
apply(KIHS_2016_pov, 2, sumNA)
```

В файле бедности есть 119 пропущенных данных для многих переменных, которые берутся из формы рабочей силы. Более того, есть некоторые переменные, такие как TotLand и OwnLand, в которых много пропущенных записей, поскольку ответов на некоторые вопросы не было (например, домохозяйства, проживающие в городских районах, не имеют какого-либо участка земли).

Это доказательство предполагает выбор только переменных, необходимых для прогнозирования, избегая нежелательного условия отбрасывания всех строк, в которых присутствует какое-либо пропущенное значение. Этот выбор будет сделан в следующем подразделе.

9.3.4 Импорт и обработка других индивидуальных характеристик (f1_nal.sav)

Важность импорта этого набора данных зависит от необходимости учитывать уровень образования всех членов домохозяйства, а не только уровень главы домохозяйства. Файл f1_nal.sav содержит образовательную информацию о 19 355 человек, участвовавших в опросе. Переменная c9 записывает уровень образования для всех членов домашних хозяйств. Эта переменная является категориальной переменной с 11 категориями, как это было первоначально зафиксировано в обследовании. Начиная с переменной c9, категории были свёрнуты в 3 категории: «Высшее образование» (HigherEducation); «Профессиональное образование» (Prof.Educ), которое включает в себя среднее профессиональное образование; и «Общее среднее образование или менее» (Secondaryorless), которое включает в себя начальное техническое образование. Это перекодирование включает в себя многократное (вложенное) использование команды ifelse ().

Наконец, число членов д/х с «Высшим образованием» и «Профессиональным образованием» было подсчитано для каждого домохозяйства в выборке. Для этой процедуры требуется команда tapply () и небольшая запись в коде, описанная во Вставке из R ниже.

В конце были созданы две новые переменные на уровне домохозяйств, представляющие количество членов с высшим и профессиональным образованием соответственно. Эти переменные были добавлены в файл бедности.

Вставка из R9-4.

Построение переменной количества членов домохозяйства по уровню образования

```
## SECOND DATASET: individual information (education)
ind_data<-as.data.frame(read_sav("f1_nal.sav"))

## recoding education
ind_data$educ<-ifelse(ind_data$c9==1, "1 Higher",
  ifelse(ind_data$c9==2, "2 Incomplete Higher",
    ifelse(ind_data$c9==3, "3 Secondary Prof.",
      ifelse(ind_data$c9==5, "5 Genaral Secondary (complete)",
        ifelse(ind_data$c9==6, "6 Genaral Secondary (incomplete)",
          ifelse(ind_data$c9==7, "7 Elementary",
            ifelse(ind_data$c9==8, "8 No elementary",
              ifelse(ind_data$c9==9, "9 Illiterate",
                ifelse(ind_data$c9==41, "41 Primary Prof. Technical (with Genaral Secondary)",
                  ifelse(ind_data$c9==42, "42 Primary Prof. Technical (without Genaral Secondary)",
                    "Children of 0-5 years old")))))))))))

ind_data$Educ4<-ifelse(ind_data$educ== "1 Higher","Higher Educ",
  ifelse(ind_data$educ== "2 Incomplete Higher"|
ind_data$educ== "3 Secondary Prof.", "Prof. Educ",
  "Secondary or less"))
```

```

LeUni(ind_data$hh_code)

## [1] 5015

ind_data$HigherEduc<-ifelse(ind_data$Educ4=="Higher Educ", 1, 0)
ind_data$ProfEduc<-ifelse(ind_data$Educ4=="Prof. Educ", 1, 0)

KIH5_2016_pov$NHighEduc<-tapply(ind_data[, "HigherEduc"], ind_data[, "hh_code"], sum)
KIH5_2016_pov$NProfEduc<-tapply(ind_data[, "ProfEduc"], ind_data[, "hh_code"], sum)

```



Применение дополнительной функции `LeUni()` полезно в этом контексте, чтобы проверить, все ли домохозяйства были рассмотрены в отдельном файле.

Файл бедности теперь имеет ещё две переменные и размерность 5015 строк и 172 столбцов. Как указывалось ранее, в некоторых столбцах много пропущенных записей (даже если в них не обязательно пропущены данные). Поэтому были выбраны только те переменные, которые строго необходимы для прогнозирования.

Конечно, 119 пропущенных данных из формы рабочей силы по-прежнему отсутствуют, поскольку они присутствуют в переменных `Nempl` и `hhempl`. Для подсчёта количества пропущенных значений в каждом столбце применяется дополнительная функция `sumNA()`. Файл домашнего хозяйства был назван `KIH5_2016_pov`.

Вставка из R9-5. Выбор характеристик домохозяйств

```

## Selection of only needed variables to avoid missing entries of variables we are not
## interested in
##The list of selected variables of the poverty file
selected.variables<-c("hh_code", "weight", "expfact", "year", "inc2", "oblast", "b002",
"hhhage", "hheduc", "hhempl", "hhsex", "hsize", "Nempl",
"NHighEduc", "NProfEduc", "pccd", "pline", "cpsc")

KIH5_2016_pov<-KIH5_2016_pov[, selected.variables]
apply(KIH5_2016_pov, 2, sumNA)

## hh_code weight expfact year inc2 oblast b002
## 0 0 0 0 0 0 0
## hhhage hheduc hhempl hhsex hsize Nempl NHighEduc
## 0 0 119 0 0 119 0
## NProfEduc pccd pline cpsc
## 0 0 0 0

```

9.3.5 Импорт и обработка характеристик жилищных условий(f7_01.sav)

Набор данных `f7_01.sav` относится к Разделу 7.1 вопросника, в котором рассматриваются условия домохозяйств, такие как тип жилья, количество комнат в домохозяйстве, наличие канализации и тому подобное. Этот вопросник был передан в д/х выборки в третьем квартале.

Этот набор данных имеет размерность 4977 строк и 100 столбцов: 38 домохозяйств не были опрошены в третьем квартале и, следовательно, их нет в этом наборе данных.

В этом наборе данных во многих столбцах отсутствуют записи, поскольку домохозяйства не имеют права отвечать. Например, только три домохозяйства заявили, что имеют дачу (вопрос `q41_a4`), а остальные 4974 значения отсутствуют. Поэтому обязательно выбирать только те переменные, которые являются информативными по условиям домохозяйства. Этот набор данных был назван `KIH5_2016_f7_01`.

The selection can be done using the `subset()` command in R and yield 18 variables as potential predictors.

Вставка из R9-6. Импорт и обработка файла жилищных условий

```
## THIRD DATASET: The housing conditions part of the questionnaire (form 7 section 1)
KIHS_2016_f7_01<-as.data.frame(read_sav("f7_01.sav"))
dim(KIHS_2016_f7_01)
## [1] 4977 100
##dim=4977 since only 4973 hh were interviewed in the 3rd quarter

## Selecting the variables
KIHS_2016_f7_01<-subset(KIHS_2016_f7_01,
select=c(hh_code, q1, q3, q4, q5, q6, q7, q14, q16r1c2,
          q16r2c2, q16r3c2, q16r4c2, q16r5c2, q16r6c2, q16r7c2,
          q16r8c2, q16r9c2, q16r10c2))

names(KIHS_2016_f7_01)<-c("hh_code", "type_accommodation", "housing_ownership",
"getting_accommodation", "total_area_sm", "living_area_sm", "number_of_rooms",
"walls_material", "centralized_heating", "individual_heating_system",
"water_pipes", "sewage", "hot_water_supply", "centralized_gas_supply",
"bath_or_shower",
"telephone", "electric_stove", "electricity")

##Checking for missing data
apply(KIHS_2016_f7_01, 2, sumNA)

##           hh_code           type_accommodation
##           0              0
## housing_ownership getting_accommodation
##           0              0
## total_area_sm      living_area_sm
##           0              0
## number_of_rooms    walls_material
##           0              0
## centralized_heating individual_heating_system
##           0              0
## water_pipes        sewage
##           0              0
## hot_water_supply   centralized_gas_supply
##           0              0
## bath_or_shower     telephone
##           0              0
## electric_stove     electricity
##           0              0
```

Проверка отсутствующих данных, в результате, приводит к полному набору данных.

9.3.6 Импорт и обработка наличия товаров длительного пользования (f7_02.sav)

Набор данных f7_02.sav ссылается на Раздел 7.2 вопросника, в котором рассматривается наличие 49 товаров длительного пользования и соответствующей информации (например, год покупки).

Размер файла составляет 57338 строк и 6 столбцов. Количество строк отражает...

Каждый ряд представляет конкретный предмет длительного пользования, имеющийся в домашнем хозяйстве. Таким образом, одно и то же домашнее хозяйство регистрируется столько раз, сколько имеется товаров длительного пользования. Например, домашнее хозяйство 20001 имеет 8 записей (строк) в наборе данных, поскольку оно имеет цветной телевизор (кодировое обозначение № 6), утюг (кодировое обозначение № 7), видеомэгафитон (кодировое обозначение № 9) и так далее ...

Столбцы набора данных представляют, помимо кода домохозяйства, код товара длительного пользования (с1), количество (с2), год производства (с3), год покупки (с4) и приблизительную стоимость товара длительного пользования. (с5), точно в соответствии с формой вопросника (см. Раздел 7.2 вопросника).

Перед объединением данных необходимо также упорядочить последний набор данных аналогично другим, что означает, что каждая строка должна представлять домохозяйство, а столбцы должны указывать наличие в собственности (или отсутствие) каждого конкретного предмета длительного пользования.

В анкетезадаются вопросы по 49 товарам длительного пользования, но не все товары длительного пользования потенциально могут повлиять на статус бедности. На основании связи между уровнем бедности и собственностью и, отзывов местных учреждений было выбрано 11 из 49 товаров длительного пользования с использованием дополнительной функции `Select ()`.

Вставка из R9-7. Импорт и обработка файла наличия товаров длительного пользования

```
## FOURTHDATASET: Availability of durables (form 7 section 2)
KHS_2016_f7_02<-as.data.frame(read_sav("f7_02.sav"))
KHS_2016_f7_02[1:5,]

##   hh_code c1 c2  c3  c4  c5
## 1   20001  6  1 2012 2012 5000
## 2   20001  7  1 2001 2001 1000
## 3   20001  9  1 2012 2012 1800
## 4   20001 34  1 2010 2010 7000
## 5   20001 40  1 2010 2010 3000

dim(KHS_2016_f7_02)

## [1] 57338      6

LeUni(KHS_2016_f7_02$hh_code)

## [1] 4977

Durables<-Select(KHS_2016_f7_02[, c("hh_code", "c1")],
Codes=c(60, 34, 6, 58, 15, 25, 19, 40, 31, 32, 33),
Names=c("SatAntenna", "ElOven", "TVCol", "LandLine", "PC", "Car",
"WshMachine", "Sofa", "Ref1", "Ref2", "Freezer"))

## The durable-level durables dataset
Durables[1:4,]

##   hh_code c1 SatAntenna ElOven TVCol LandLine PC Car WshMachine Sofa Ref1
## 1   20001  6          0      0      1      0  0  0          0  0  0
## 2   20001  7          0      0      0      0  0  0          0  0  0
## 3   20001  9          0      0      0      0  0  0          0  0  0
## 4   20001 34          0      1      0      0  0  0          0  0  0
##   Ref2 Freezer
## 1      0      0
## 2      0      0
## 3      0      0
## 4      0      0

dim(Durables)
## [1] 57338     13
```

Этот файл с названием `Durables` по-прежнему содержит исходное число строк и 13 столбцов, соответствующих коду домохозяйства, коду товара длительного

пользования и его наличием или отсутствием (то есть двоичной переменной) для каждого из 11 выбранных товаров длительного пользования.



Обратите внимание, что выбор товаров длительного пользования может быть легко изменён внутри функции `Select ()`, изменяя (добавляя / удаляя) код товара длительного пользования, и соответствующего названия в том же порядке. Это возможно, поскольку `Codes=c()` и `Names=c()` являются аргументами функции. Введите `?args` в консоли R для подробностей об аргументах функции.

Последний этап - агрегирование файла `Durables` (товары длительного пользования) на уровне домохозяйства с помощью дополнительной функции `Aggr ()`, которая зависит от предыдущего файла с названием `Durables` и названий столбцов нового набора данных, которые являются аргументами функции. Набор столбцов содержит код домашнего хозяйства и выбранные товары длительного пользования. Применение этой функции к набору данных `Durables` даёт набор данных, где домохозяйства указаны в строках. Этот файл на уровне домашнего хозяйства называется `Durables_hh` с размером 4977 строк и 12 столбцов (`hh_code` плюс 11 товаров длительного пользования).

Вставка из R9-8. Построение набора данных о наличии товаров длительного пользования на уровне домашних хозяйств.

```
## The household dataset for durables
Names=c("hh_code", "SatAntenna", "ElOven", "TVCol", "LandLine", "PC", "Car",
"WshMachine", "Sofa", "Ref1", "Ref2", "Freezer")
Durables_hh<-Aggr(Durables, Names=Names)

Durables_hh[1:4,]

##   hh_code SatAntenna ElOven TVCol LandLine PC Car WshMachine Sofa Ref1
## 1  20001         1      1     1      0  0  0          0     1     0
## 2  20002         0      0     1      0  0  0          0     0     1
## 3  20003         0      0     1      0  0  1          0     1     0
## 4  20004         1      0     1      0  0  0          0     0     1
##   Ref2 Freezer
## 1     0       0
## 2     0       0
## 3     0       0
## 4     0       0

dim(Durables_hh)

## [1] 4977  12

apply(Durables_hh, 2, sumNA)

##   hh_code SatAntenna ElOven TVCol LandLine PC
##      0         0      0     0      0     0
##   Car WshMachine Sofa Ref1 Ref2 Freezer
##      0         0      0     0     0     0
```

9.3.7 Объединение наборов данных и сохранение данных

После того, как наборы данных были импортированы и очищены, их необходимо объединить для работы только с одним файлом, в котором строки представляют домохозяйства, а столбцы - выбранные потенциальные предикторы состояния бедности.

Команда «слияние» - `merge ()` в R объединяет соответствующие наблюдения (домохозяйства) из набора данных бедности с наблюдениями из набора данных о

жилищных условиях и из набора данных о наличии товаров длительного пользования, последовательно сопоставляя файлы по коду домохозяйства (hh_code). Ввод Merge в консоли RStudio откроет окно с подробной информацией о команде и о том, как её использовать.

Окончательный набор данных с названием KINS_2016 можно сохранить в виде файла SPSS с помощью команды write_sav (), которая находится в библиотечном хранилище libraryheaven.

Вставка из R9-9. Слияние наборов данных

```
##Merging the datasets
KINS_2016<-na.omit(merge(merge(KINS_2016_pov, KINS_2016_f7_01, by="hh_code"),
                           Durables_hh, by="hh_code"))

dim(KINS_2016)
## [1] 4889  46

##Exporting it as a spss file
write_sav(KINS_2016, "KINS_2016.sav")
```



Файл бедности (KINS_2016_pov) по-прежнему содержит 119 недостающих данных. Для удаления строк с отсутствующими данными в процедуре объединения была использована функция na.omit () (см. Вставка из R 9-9).

Полный окончательный файл (KINS_2016) имеет размер 4889 строк и 46 столбцов.

В общей сложности 126 домашних хозяйств отсутствуют в первоначальной выборке из 5015 единиц. Некоторые из них пропущены из-за отсутствия значений в переменных рабочей силы, а некоторые - из-за отсутствия ответа в Форме 7.1 и 7.2 опросника. Число 126 отличается от суммы пропущенных значений в отдельных файлах (119 + 38), поскольку 31 домохозяйство не ответило во всех трёх формах.

9.4 Приложение 4: Файлы кода R

Все этапы кода внутри каждого файла кода R (сценарий, или скрипт на языке R) могут быть запущены все сразу командой `source()`, которая должна быть применена к сценарию. Например, если пользователь хочет запустить файл сценария с названием `data_2016_NSC.R` всего за один шаг, достаточно ввести в консоли RStudio: `source("C: / НАЗВАНИЕ ВАШЕГО КАТАЛОГА / data_2016_NSC.R")`. Это особенно полезно, когда есть много функций, которые можно поместить только в один файл `.R`, например, дополнительные функции в начале скрипта `data_2016_NSC.R`.

В качестве альтернативы пользователь может запускать код шаг за шагом, используя кнопку «Выполнить» - Run на панели инструментов (см. Рисунок 9-3).

Далее сообщается обо всех необработанных файлах скрипта (кода).

9.4.1 Functions.R: дополнительные функции для создания первичного набора данных

```
## Importing the extra functions for creating the primary dataset
## Filename=Functions.R

##Basic functions
LeUni <- function(vec) {
  length(unique(vec))
} ##End of the function

sumNA<-function(vec){
  sum(is.na(vec))
} ##End of the function

### COLLAPSING THIS DATASET TO HH
## Taking the most important variables, selecting the column indicating the possession and finally
aggregating for households
## a) selection of some variables (60=Satellite antenna, 34==Electric oven and so on)

Select<-function(dat, Codes=c(60, 34, 6, 58), Names=c("SatAntenna", "ElOven", "TVCol", "LandLine"))
## The selection function
{
  ## dat The person-level data
  ## Codes The numbers of selected durables
  ##. Names The names of the new columns

  if(length(Codes)!=length(Names)){
    print("STOP: they should have same length")
  }

  for(i in 1: length(Codes))
  {

    dat[, Names[i]]<-ifelse(dat[, "c1"]==Codes[i], 1, 0)

  }
  return(dat)
}

##Aggregation---from durables to households
Aggr<-function(dat, Names=c("hh_code", "SatAntenna", "ElOven", "TVCol", "LandLine"))
{

  dat_hh<-as.data.frame(matrix(0, nrow=LeUni(dat[, "hh_code"]), ncol=length(Names)))
  names(dat_hh)<-Names

  for(i in 1:length(Names))
  {
    dat_hh[,i]<-tapply(dat[, Names[i]], dat[, "hh_code"], max)
  }

  return(dat_hh)
}
```

9.4.2 data_2016_NSC.R: базовый скрипт для создания первичного набора данных для министерства экономики

```
## Importing the datasets for preparation of the final dataset to MoE for forecasting
## Aimed to NSC
## Data 2016
## Filename=data_2016_NSC.R

#####
##Importing the extra functions (in the file Functions.R)
#####
source("C:/YOUR DIRECTORY/Functions.R")
ls()

#####
##Importing the library haven for reading SPSS files
#####
options(repos = c(CRAN = "http://cran.rstudio.com"))
# requiring packages
ipak<- function(pkg){
  new.pkg <- pkg[!(pkg %in% installed.packages()[, "Package"])]
  if (length(new.pkg))
    install.packages(new.pkg, dependencies = TRUE)
  sapply(pkg, require, character.only = TRUE)
} ##end of requiring packages function

packages<- c("haven")
ipak(packages)

#####
##Setting the working directory
#####
## Make sure that RStudio is in the directory where the data are
## The command getwd() will give the directory where you are

setwd("C:/YOUR DIRECTORY/KIHS_2016_SPSS")

#####
##Importing the first dataset ("PROFIL_2016.sav")
#####
## Reading the file in SPSS--as given from National Statistical Office
## FIRST DATASET: basic characteristics of the households
KIHS_2016_pov<-as.data.frame(read_sav("PROFIL_2016.sav"))
KIHS_2016_pov[1:4,]

## Looking for missing data
apply(KIHS_2016_pov, 2, sumNA)
## 119 missing from the labor force form but many for some variables (for example TotLand==1151,
OwnLand==1270)

#####
##Importing the second dataset ("f1_nal.sav")
#####
##SECOND DATASET: individual characteristics (education)
ind_data<-as.data.frame(read_sav("f1_nal.sav"))

## recoding education
ind_data$educ<-ifelse(ind_data$c9==1, "1 Higher",
  ifelse(ind_data$c9==2, "2 Incomplete Higher",
    ifelse(ind_data$c9==3, "3 Secondary Prof.",
      ifelse(ind_data$c9==5, "5 Genaral Secondary (complete)",
        ifelse(ind_data$c9==6, "6 Genaral Secondary (incomplete)",
          ifelse(ind_data$c9==7, "7 Elementary",
            ifelse(ind_data$c9==8, "8 No elementary",
              ifelse(ind_data$c9==9, "9 Illiterate",
                ifelse(ind_data$c9==41, "41 Primary Prof. Technical (with Genaral Secondary)",
                  ifelse(ind_data$c9==42, "42 Primary Prof. Technical (without Genaral Secondary)",
                    "Children of 0-5 years old")))))))))))

ind_data[1:5,]
table(ind_data$educ)

ind_data$Educ4<-ifelse(ind_data$educ== "1 Higher","Higher Educ",
  ifelse (ind_data$educ== "2 Incomplete Higher" |
    ind_data$educ== "3 Secondary Prof.", "Prof. Educ",
```

```

"Secondary or less"))

table(ind_data$Educ4)
LeUni(ind_data$hh_code)
ind_data$HigherEduc<-ifelse(ind_data$Educ4=="Higher Educ", 1, 0)
ind_data$ProfEduc<-ifelse(ind_data$Educ4=="Prof. Educ", 1, 0)

KIHS_2016_pov$NHighEduc<-tapply(ind_data[, "HigherEduc"], ind_data[, "hh_code"], sum)
KIHS_2016_pov$NProfEduc<-tapply(ind_data[, "ProfEduc"], ind_data[, "hh_code"], sum)
KIHS_2016_pov[1:4,]

## Selection of only needed variables to avoid missing entries of variables we are not interested in
##the list of selected variables of the poverty file
selected.variables<-c("hh_code", "weight", "expfact", "year", "inc2", "oblast", "b002", "hhhage",
"hhheduc", "hhhempl", "hhhsex", "hsize", "Nempl",
"NHHighEduc", "NProfEduc", "pccd", "pline", "cpsc")

KIHS_2016_pov<-KIHS_2016_pov[, selected.variables]
apply(KIHS_2016_pov, 2, sumNA)
KIHS_2016_pov[1:4,]

#####
##Importing the third dataset ("f7_01.sav.sav")
#####
## THIRD DATASET: The housing conditions part of the questionnaire (form 7 section 1)
KIHS_2016_f7_01<-as.data.frame(read_sav("f7_01.sav"))
KIHS_2016_f7_01[1:5,]
dim(KIHS_2016_f7_01)

##dim=4977 since only 4973 hh were interviewed in the 3rd quarter
## Selecting the variables not missing
KIHS_2016_f7_01<-subset(KIHS_2016_f7_01,
select=c(hh_code, q1, q3, q4, q5, q6, q7, q14, q16r1c2,
q16r2c2, q16r3c2, q16r4c2, q16r5c2, q16r6c2, q16r7c2,
q16r8c2, q16r9c2, q16r10c2))

names(KIHS_2016_f7_01)<-c("hh_code", "type_accommodation", "housing_ownership",
"getting_accommodation",
"total_area_sm", "living_area_sm", "number_of_rooms", "walls_material",
"centralized_heating", "individual_heating_system", "water_pipes", "sewage",
"hot_water_supply", "centralized_gas_supply", "bath_or_shower", "telephone",
"electric_stove", "electricity")

##Checking for missing data
apply(KIHS_2016_f7_01, 2, sumNA)
KIHS_2016_f7_01[1:5,]

#####
##Importing the fourth dataset ("f7_02.sav.sav")
#####
## FOURTH DATASET: Availability of durables (form 7 section 2)
KIHS_2016_f7_02<-as.data.frame(read_sav("f7_02.sav"))
KIHS_2016_f7_02[1:5,]
dim(KIHS_2016_f7_02)
LeUni(KIHS_2016_f7_02$hh_code)

##dim=57338 individuals equal to 4977 households.
##Some missing data and same but same of f7_01

Durables<-Select(KIHS_2016_f7_02[, c("hh_code", "c1")],
Codes=c(60, 34, 6, 58, 15, 25, 19, 40, 31, 32, 33),
Names=c("SatAntenna", "ElOven", "TVCol", "LandLine", "PC", "Car", "WshMachine",
"Sofa",
"Ref1", "Ref2", "Freezer"))

## The durable-level durables dataset
Durables[1:4,]
dim(Durables)

## The household dataset for durables
Names=c("hh_code", "SatAntenna", "ElOven", "TVCol", "LandLine", "PC", "Car", "WshMachine", "Sofa",
"Ref1", "Ref2", "Freezer")

```

```

Durables_hh<-Aggr(Durables, Names=Names)
Durables_hh[1:4,]
dim(Durables_hh)

apply(Durables_hh, 2, sumNA)
Durables[1:4,]

#####
##Merging the datasets and save the data
#####
##Merging the three datasets
KIHS_2016<-na.omit(merge(merge(KIHS_2016_pov, KIHS_2016_f7_01, by="hh_code"),
                          Durables_hh, by="hh_code"))

apply(KIHS_2016, 2, sumNA)
dim(KIHS_2016)
names(KIHS_2016)
KIHS_2016[1:5,]

##Exporting it as a spss file
write_sav(KIHS_2016, "KIHS_2016.sav")

```

9.4.3 data_2016_MoE.R: Подготовка данных

```

## Importing the dataset as received from NSC and first look at the data
## Data 2016
## Filename=data_2016_MoE.R

#####
#CRAN package repository
#####

options(repos = c(CRAN = "http://cran.rstudio.com"))

#####
#Getting the libraries
#####

# requiring packages
ipak<- function(pkg){
  new.pkg <- pkg[!(pkg %in% installed.packages()[, "Package"])]
  if (length(new.pkg))
    install.packages(new.pkg, dependencies = TRUE)
  sapply(pkg, require, character.only = TRUE)
} ##end of requiring packages function

packages<- c("haven", "rio")
ipak(packages)

#####
#The household-level dataset
#####

## Setting the working directory
setwd("C:/YOUR DIRECTORY")

## Reading the file in SPSS---as given from National Statistical Office
KIHS_2016<-as.data.frame(read_sav("KIHS_2016.sav"))
KIHS_2016[1:4,]

## Checking for missing data
sumNA<-function(vec){
  sum(is.na(vec))
}
apply(KIHS_2016, 2, sumNA)

#####
# Re-codification of potential predictors and outcome
#####
## Area of residence
KIHS_2016$Area<-ifelse(KIHS_2016$b002==1, "Urban", "Rural")

##Type of accomodation
KIHS_2016$Rooms<-ifelse(KIHS_2016$number_of_rooms>=6, 6, KIHS_2016$number_of_rooms)

```

```

##mq of the toral are of the housing
KIHS_2016$Mq<-ifelse(KIHS_2016$total_area_sm>=300, 300, KIHS_2016$total_area_sm)

##Availability of sewer services
KIHS_2016$Sewer<-ifelse(KIHS_2016$sewage==1, 1, 0)

##Availability of electric stove
KIHS_2016$ElStove<-ifelse(KIHS_2016$electric_stove==1, 1, 0)

##Size of household's size
KIHS_2016$Hsize<-ifelse(KIHS_2016$hsize>=7, 7, KIHS_2016$hsize)

##Household's head gender
KIHS_2016$hhhsex<-ifelse(KIHS_2016$hhhsex==1, "1 Male", "2 Female")

##Household's head education
KIHS_2016$Educ4<-ifelse(KIHS_2016$hhheduc== 1,"Higher Educ",
ifelse (KIHS_2016$hhheduc== 2 |
        KIHS_2016$hhheduc== 3 |
        KIHS_2016$hhheduc== 41 |
        KIHS_2016$hhheduc== 42, "Prof. Educ",
        "Secondary or less"))

KIHS_2016$Educ4<-relevel(as.factor(KIHS_2016$Educ4), ref ="Secondary or less")

##Household's head employment Status
KIHS_2016$HHempl<-ifelse(KIHS_2016$hhhempl==1, 1, 0)

##Number od employed in the HH
KIHS_2016$Nempl<-ifelse(KIHS_2016$Nempl>3, 4, KIHS_2016$Nempl)

##Giving names to oblasts
KIHS_2016$oblast<-as.factor(ifelse(KIHS_2016$oblast==41702, "41702 Issykul",
ifelse(KIHS_2016$oblast==41703, "41703 Jalal-Abad",
ifelse(KIHS_2016$oblast==41704, "41704 Naryn",
ifelse(KIHS_2016$oblast==41705, "41705 Batken",
ifelse(KIHS_2016$oblast==41706, "41706 Osh",
ifelse(KIHS_2016$oblast==41707, "41707 Talas",
ifelse(KIHS_2016$oblast==41708, "41708 Chui",
ifelse(KIHS_2016$oblast==41711, "41711 Bishkek",
"41721 Osh city")))))))))))

## Outcome variable
KIHS_2016$poor<-ifelse(KIHS_2016$cpsc==100, 1, 0)

#####
#The Oblast-level dataset
#####
all.oblast <-import("~/Dropbox/Poverty_Forecast/Data/Oblast_dataset.xls")
KIHS_2016$full.GDP <- all.oblast$GDP_2016[KIHS_2016$oblast] ## Per capita GDP at Oblast level
KIHS_2016$full.UNEMP <- all.oblast$Unemp_2016[KIHS_2016$oblast] ## Per capita GDP at Oblast level

#####
# Selection of potential predictors
#####
##Selecting the most important predictors among the available for forecasting
data_forecasting_2016<-subset(KIHS_2016, select=c("hh_code", "weight",
"expfact", "year", "inc2", "oblast", "Area", "hhhage", "hhhsex", "Educ4",
"Hsize", "Nempl", "NHighEduc", "NProfEduc", "poor", "Rooms", "ElStove",
"SatAntenna", "LandLine", "Car", "WshMachine", "Ref1", "Ref2",
"Freezer", "full.UNEMP", "full.GDP"))

#####
#Saving the file for forecasting
#####
save(data_forecasting_2016, file = "data_forecasting_2016.RData")

```

9.4.4 Checking_data_2016.R: Проверка данных

```

## Checking the data and tabulate
## Checking_data_2016.R

#####
#Loading the data
#####

```

```

load("data_forecasting_2016.RData")

#####
#Structure and summary statistics of the dataset
#####
str(data_forecasting_2016)
summary(data_forecasting_2016)

#####
#Frequency distribution
#####
table(data_forecasting_2016[, "poor"])
prop.table(table(data_forecasting_2016[, "poor"]))

mean(data_forecasting_2016[, "inc2"])
tapply(data_forecasting_2016[, "inc2"], data_forecasting_2016[, "poor"], mean)

#####
#Relationship btw poor hh and other variables
#####
table(data_forecasting_2016[, "oblast"], data_forecasting_2016[, "poor"])

#####
#The use of sampling weights
#####

## Households weights
weighted.mean(x=data_forecasting_2016[, "poor"], w=data_forecasting_2016[, "expfact"])

## Individual weights
weighted.mean(x=data_forecasting_2016[, "poor"], w=data_forecasting_2016[, "weight"])

##Weighted mean by oblast
sapply(split(data_forecasting_2016, data_forecasting_2016[, "oblast"]),
function(data_forecasting_2016) weighted.mean(x=data_forecasting_2016[, "poor"],
                                              w=data_forecasting_2016[, "weight"]))

```

9.4.5 Model_estimation_2016.R: Оценка модели

```

## Modeling poverty with a logistic-regression model in Kyrgystan using micro-variables
## Filename Model_estimation_2016.R

#####
##Importing the library arm for reading the model outcome
#####

options(repos = c(CRAN = "http://cran.rstudio.com"))

ipak<- function(pkg){
  new.pkg <- pkg[!(pkg %in% installed.packages()[, "Package"])]
  if (length(new.pkg))
    install.packages(new.pkg, dependencies = TRUE)
  sapply(pkg, require, character.only = TRUE)
}

packages<- c("arm")
ipak(packages)

#####
## Setting the working directory and loading the data with the predictors
#####
setwd("C:/YOUR DIRECTORY")
load("data_forecasting_2016.RData")

dim(data_forecasting_2016)
data_forecasting_2016[1:5, ]

#####
## Model Estimation for the year 2016
#####
lrfit<-glm(poor~ rescale(hhhage) + hhhsex + rescale(I(NEmpl/Hsize)) + rescale(I(NHighEduc/Hsize)) +
rescale(I(NProfEduc/Hsize)) +
rescale(I(inc2/Hsize)) + factor(Rooms) + factor(Hsize) + rescale(full.GDP)+
rescale(full.UNEMP) + Area +
SatAntenna+ LandLine +
Car+ ElStove + WshMachine + Ref1 + Ref2+

```



```

        Area * rescale(full.UNEMP),
family=binomial(link = "logit"), data=data_forecasting_2016)

#####
## Assessing the role of the predictors
#####
display(lrfit,3)

#####
## Evaluation and accuracy of the model
#####
## Residuals
##Estimated values
pred<- lrfit$fitted.values
##Observed values
y <- data_forecasting_2016$poor
##residuals
res<- y-pred

#####
## The binned residuals plot
#####
binnedplot(pred ,res, nclass=60,
xlab="Expected Values", ylab="Average residual",
main="Binned residual plot")

#####
## The confusion matrix and the error rate
#####
## The confusion matrix
table(y.oss=y, y.tilde=round(pred))

##error rate
mean(pred > 0.5 & y==0) + mean(pred <0.5 & y==1)

```

9.4.6 Forecasting_model_2016.R: Прогнозирование бедности

```

## Forecasting poverty with a logistic-regression model in Kyrgystan using micro-variables
## Filename Forecasting_model_2016.R

#####
##Importing the library arm and rio
#####
options(repos = c(CRAN = "http://cran.rstudio.com"))

ipak<- function(pkg){
  new.pkg <- pkg[!(pkg %in% installed.packages()[, "Package"])]
  if (length(new.pkg))
  install.packages(new.pkg, dependencies = TRUE)
  sapply(pkg, require, character.only = TRUE)
}

packages<- c("arm", "rio")
ipak(packages)

#####
## Setting the working directory and loading the data with the predictors
#####
setwd("C:/YOUR DIRECTORY")
load("data_forecasting_2016.RData")

dim(data_forecasting_2016)
data_forecasting_2016[1:5, ]

#####
## The forecasting model
#####
lrfit.forecast<-glm(poor~ hhhage + hhhsex + I(NEmpl/Hsize) + I(NHighEduc/Hsize) + I(NProfEduc/Hsize) +
I(inc2/Hsize) + factor(Rooms) + factor(Hsize) + full.GDP + full.UNEMP + Area +
SatAntenna+ LandLine +
Car+ ElStove + WshMachine + Ref1 + Ref2+
Area * full.UNEMP,
family=binomial(link = "logit"), data=data_forecasting_2016)
display(lrfit.forecast)

#####

```

```

## IN-SAMPLE FORECASTING
## The estimated poverty rate in 2016
#####
EstimatedPoverty<-lrfit.forecast$fitted.values
EstimatedPovertyRate<-weighted.mean(EstimatedPoverty, data_forecasting_2016$weight)
EstimatedPovertyRate

#####
## Estimation by Oblast
#####
data_Oblast<-data.frame(oblast=data_forecasting_2016$oblast, EstimatedPoverty,
                        weights=data_forecasting_2016$weight)
pred_Oblast<-sapply(split(data_Oblast, data_Oblast[, "oblast"]),
function(data_Oblast) weighted.mean(x=data_Oblast[, "EstimatedPoverty"],
w=data_Oblast[, "Weights"]))
pred_Oblast

#####
## Combining Oblast observed poverty rates with estimated poverty rates by Oblast
## Oblast poverty rates using the raw data
#####
observed_poverty_rates<-sapply(split(data_forecasting_2016, data_forecasting_2016[, "oblast"]),
function(data_forecasting_2016) weighted.mean(x=data_forecasting_2016[, "poor"],
w=data_forecasting_2016[, "weight"]))
Comparing_poverty_rates<-cbind(EstimatedPovRts=round(pred_Oblast*100, 1),
ObservedPovRts=round(observed_poverty_rates*100, 1))
Comparing_poverty_rates
MAE<-sum(abs(Comparing_poverty_rates[,1]-Comparing_poverty_rates[,2]))/9
MAE

#####
## OUT-OF-SAMPLE FORECASTING
#####
## Importing the MoE (August 2018) forecasting macro-variables file
macro_var <-import("C:/YOUR DIRECTORY/Oblast_dataset.xls")

#####
## Forecasting the poverty rate in 2017
#####
#The matrix of predictors for forecasting--year 2017
pred.2017<-data.frame(hhhage=data_forecasting_2016$hhhage, hhhsex=data_forecasting_2016$hhhsex,
NEmpl=data_forecasting_2016$NEmpl,
NHighEduc=data_forecasting_2016$NHighEduc,
NProfEduc=data_forecasting_2016$NProfEduc,
inc2=data_forecasting_2016$inc2,
Rooms=data_forecasting_2016$Rooms, Hsize=data_forecasting_2016$Hsize,
full.GDP=macro_var$GDP_2017[data_forecasting_2016$oblast],
full.UNEMP=macro_var$Unemp_2017[data_forecasting_2016$oblast],
Area=data_forecasting_2016$Area,
SatAntenna = data_forecasting_2016$SatAntenna,
LandLine=data_forecasting_2016$LandLine, Car=data_forecasting_2016$Car,
ElStove=data_forecasting_2016$ElStove,
WshMachine=data_forecasting_2016$WshMachine,
Ref1=data_forecasting_2016$Ref1, Ref2=data_forecasting_2016$Ref2)

#####
## Prediction of poverty status in 2017
#####
predictions_2017<-predict.glm(lrfit.forecast, pred.2017, type = "response")

#####
## Prediction of national poverty rate in 2017
#####
Est.PovertyRate_2017<-weighted.mean(predictions_2017, data_forecasting_2016$weight)
Est.PovertyRate_2017

#####
## Prediction of national poverty rate in 2017 at regional (oblast) level
#####
## Prediction pf poverty rates in 2017 at regional (oblast) level
data_Oblast$predictions_2017=predictions_2017
pred_Oblast_2017<-sapply(split(data_Oblast, data_Oblast[, "oblast"]),
function(data_Oblast)
weighted.mean(x=data_Oblast[, "predictions_2017"], w=data_Oblast[, "Weights"]))

forecast_2017<-as.matrix(round(pred_Oblast_2017*100, 2), 9, 1)
colnames(forecast_2017)="Estimated values 2017"

```

```

forecast_2017

#####
## Forecasting the poverty rate in 2018
#####
#The matrix of predictors for forecasting--year 2018
pred.2018<-data.frame(hhhage=data_forecasting_2016$hhhage, hhhsex=data_forecasting_2016$hhhsex,
                      NEmpl=data_forecasting_2016$NEmpl,
                      NHighEduc=data_forecasting_2016$NHighEduc,
NProfEduc=data_forecasting_2016$NProfEduc,
                      inc2=data_forecasting_2016$inc2,
                      Rooms=data_forecasting_2016$Rooms, Hsize=data_forecasting_2016$Hsize,
                      full.GDP=macro_var$GDP_2018[data_forecasting_2016$oblast],
                      full.UNEMP=macro_var$Unemp_2018[data_forecasting_2016$oblast],
                      Area=data_forecasting_2016$Area,
                      SatAntenna = data_forecasting_2016$SatAntenna,
                      Landline=data_forecasting_2016$Landline, Car=data_forecasting_2016$Car,
                      ElStove=data_forecasting_2016$ElStove,
                      WshMachine=data_forecasting_2016$WshMachine,
                      Ref1=data_forecasting_2016$Ref1, Ref2=data_forecasting_2016$Ref2)

#####
## Prediction of poverty status in 2018
#####
predictions_2018<-predict.glm(lrfit.forecast, pred.2018, type = "response")

#####
## Prediction of national poverty rate in 2018
#####
Est.PovertyRate_2018<-weighted.mean(predictions_2018, data_forecasting_2016$weight)
Est.PovertyRate_2018

#####
## Prediction of national poverty rate in 2018 at regional (oblast) level
#####
data_Oblast$predictions_2018=predictions_2018
pred_Oblast_2018<-sapply(split(data_Oblast, data_Oblast[, "oblast"]),
function(data_Oblast)
weighted.mean(x=data_Oblast[, "predictions_2018"], w=data_Oblast[, "Weights"]))

forecast_2018<-as.matrix(round(pred_Oblast_2018*100, 2), 9, 1)
colnames(forecast_2018)="Estimated values 2018"
forecast_2018

#####
## Forecasting the poverty rate in 2019 (ONLY MACRO-LEVEL)
#####
#The matrix of predictors for forecasting--year 2019
pred.2019<-data.frame(hhhage=data_forecasting_2016$hhhage, hhhsex=data_forecasting_2016$hhhsex,
                      NEmpl=data_forecasting_2016$NEmpl,
                      NHighEduc=data_forecasting_2016$NHighEduc,
NProfEduc=data_forecasting_2016$NProfEduc,
                      inc2=data_forecasting_2016$inc2,
                      Rooms=data_forecasting_2016$Rooms, Hsize=data_forecasting_2016$Hsize,
                      full.GDP=macro_var$GDP_2019[data_forecasting_2016$oblast],
                      full.UNEMP=macro_var$Unemp_2019[data_forecasting_2016$oblast],
                      Area=data_forecasting_2016$Area,
                      SatAntenna = data_forecasting_2016$SatAntenna,
                      Landline=data_forecasting_2016$Landline, Car=data_forecasting_2016$Car,
                      ElStove=data_forecasting_2016$ElStove,
                      WshMachine=data_forecasting_2016$WshMachine,
                      Ref1=data_forecasting_2016$Ref1, Ref2=data_forecasting_2016$Ref2)

#####
## Prediction of poverty status in 2019
#####
predictions_2019<-predict.glm(lrfit.forecast, pred.2019, type = "response")

#####
## Prediction of national poverty rate in 2019
#####
Est.PovertyRate_2019<-weighted.mean(predictions_2019, data_forecasting_2016$weight)
Est.PovertyRate_2019

#####
## Combining results at national level

```

```
#####
National_PovRates<-cbind(EstimatedPovRate2016=round(EstimatedPovertyRate*100, 2),
                        EstimatedPovRate2017=round(Est.PovertyRate_2017*100, 2),
                        EstimatedPovRate2018=round(Est.PovertyRate_2018*100, 2),
                        EstimatedPovRate2019=round(Est.PovertyRate_2019*100, 2))

colnames(National_PovRates)<-c("PR_2016", "PR_2017", "PR_2018", "PR_2019")
National_PovRates

#####
## Saving the results as an excel(.csv) file
#####
write.csv(National_PovRates, "National_PovRates.csv")

#####
## Combining results at regional level
#####
Regional_PovRates<-cbind(EstimatedPovRts2016=round(pred_Oblast*100, 2),
                        EstimatedPovRts2017=forecast_2017, forecast_2018)

colnames(Regional_PovRates)<-c("PR_2016", "PR_2017", "PR_2018")
Regional_PovRates

#####
## Saving the results as an excel(.csv) file
#####
write.csv(Regional_PovRates, "Regional_PovRates.csv")
```