aboba

Содержание

Введение

В наше время обработка данных является одним из самых востребованных областей IT области. Методы позволяют находить зависимости между различными показателями при обработке данных.

Эта работка направлена на применение полученых при обучениии навыков анализа. В качестве датасета выступает Stat100_Fall2018_Survey02 за авторством Университета Иллинойса.

Целью работы является формирование двух моделей линейной регрессии и их сравнение по ряду параметров.

На основании цели были сформулированы следующие задачи:

а) Загрузка и очистка датасета .

б) Сформировать две модели линейной регрессии (базовую и расширенную).

в) Построить визуализацю для каждой модели .

г) Сделать выводы на основе полученных метрик и характеристик

1 Подготовка данных

Для начала работу требуется загрузить найденный датасет. Но заголовки, заданные по умолчанию, содержать поясняющую информацию. Поэтому требуется отдельно загрузить данные и их заголовки. Заранее все заголовки были перенесены в файд headers.txt. Загрузка основной информации происходить с помощью кода, представленного в листинге 1.1.

Листинг 1.1 — Считывание датасета

```
setwd('/home/ndavs/study/math/DS/r/5_lab_Stats_100')

data <- read.table('data.dat', sep=' ')
dim(data)
head(data)
```

Данные о размерности: (1458 ,35) Первые пять записей представлены в таблице 1.1

Таблица 1.1 — Сводная таблица наблюдений: первые и последние 5 переменных (остальные обозначены как $\cdots$)

Obs	V1	V2	V3	V4	V5	$\cdots$	V31	V32	V33	V34	V35
Obs1	1	1	1	0	2	$\cdots$	0	100	7	1	NA
Obs2	2	1	1	0	3	$\cdots$	0	100	9	1	NA
Obs3	3	1	1	1	2	$\cdots$	0	100	3	1	NA
Obs4	4	1	1	0	2	$\cdots$	15	0	1	1	NA
Obs5	5	0	0	1	2	$\cdots$	0	50	5	1	NA

Для обработки заголовки был использован следующий подход (листинг 1.2):

— считывание строки с заголовками;

— Использование регулярного выражения для удаления скобок и значений в них;

— Удаление лишних пробелов;

— Разделение заначений на элементы массива

Листинг 1.2 — Работа с заголовками

```
base_headers <- readLines('headers.txt', n=1)
cleaned_header <- gsub("\\s*\\([^()]*\\)", "", base_headers)

```

```
cleaned_header <- gsub("\\s+", " ", cleaned_header)
cleaned_header <- trimws(cleaned_header)
cleaned_header <- unlist(strsplit(cleaned_header, " "))
length(cleaned_header)
```

Длина полученного массива: 34 (на один меньше, чем количество столбцов датафрейма, т.к. последние значения там NULL)

После проделаных шагов, значения заголовков добавляются к датасету (листинг 1.3).

Листинг 1.3 — Применение заголовков

```
colnames(data) <- cleaned_header
dim(data)
head(data)
```

Размерность не изменилась: (1458, 35).

Первые пять строк полученного датасета представлены в таб. 1.2.

Таблица 1.2 — Социально-академические характеристики студентов: только первые и последние 4 переменных

№	Gender	Gender_ID	Greek	Home_Town	···	Work_Hours	Tuition	Career	Section
1	1	1	1	0	···	0	100	7	1
2	2	1	1	0	···	0	100	9	1
3	3	1	1	1	···	0	100	3	1
4	4	1	1	0	···	15	0	1	1
5	5	0	0	1	···	0	50	5	1
6	6	1	1	0	···	0	90	5	1

Далее подготовим только нужные значения из датайрейма(1.4).

Листинг 1.4 — Сформируем датафрейм нужных признаков

```
dd <- data[, c("Drinks_per_week", "Party_Hours_per_week", "Gender",
    "Home_Town")]
head(dd)
dim(dd)
summary(dd)
```

Описание данных:

а) **Пол (Gender)**:
   — 0 = Мужской
   — 1 = Женский

б) **Родной город (Home_Town)**: тип населенного пункта:

— 0 = Маленький город

— 1 = Средний город

— 2 = Большой город (пригород)

— 3 = Большой город (без пригородов)

в) **Часы вечеринок в неделю (Party_Hours_per_week)**: среднее количество часов, проведенных на вечеринках в неделю.

г) **Алкогольные напитки в неделю (Drinks_per_week)**: среднее количество алкогольных напитков, потребляемых за неделю.

Преобразуем параметр города следующим образом: 0=Small Town+Medium City, 1=Big City. Также пол и города преобразуем в факторную переменную (листинг 1.5).

Листинг 1.5 — Преобразование нужных признаков

```
dd$Home_Town[dd$Home_Town < 2] <- 0
dd$Home_Town[dd$Home_Town >= 2] <- 1
dd$Gender <- factor(dd$Gender)
```

Первые пять строк полученного датасета представлены в таблице 1.3

Таблица 1.3 — Параметры студентов, связанные с поведением и происхождением

№	Drinks_per_week	Party_Hours_per_week	Gender	Home_Town
1	1	1	1	1
2	0	0	1	1
3	10	6	1	1
4	35	20	1	1
5	25	12	0	1
6	3	8	1	1

Размерность полученного датафрейма: (1458, 4).

Сводная информаци по полученному датафрейму представлена в таб. 1.4.

Разделим полученый датафрейм на обучающую, валидационную и тестовую выборки (листинг 1.6)

Таблица 1.4 — Описательная статистика по ключевым переменным

Показатель	Drinks_per_week	Party_Hs_per_wk	Gender	Home_Tn
Min	0.000	0.000	0.0000	0.000
1st Quartile	0.000	1.000	0.0000	1.000
Median	3.000	4.000	1.0000	2.000
Mean	6.475	5.496	0.6454	1.853
3rd Quartile	10.000	8.000	1.0000	3.000
Max	50.000	50.000	1.0000	3.000

Листинг 1.6 — Разделение на выборки

```
library(caret)

set.seed(2004)

train_index <- caret::createDataPartition(dd$Drinks_per_week, p = 0.7,
    list = FALSE)
train_data <- dd[train_index, ]
temp_data <- dd[-train_index, ]

validation_index <- createDataPartition(temp_data$Drinks_per_week, p =
    0.5, list = FALSE)
validation_data <- temp_data[validation_index, ]
test_data <- temp_data[-validation_index, ]
```

На этом подготовка данных завершена.

2 Создание моделей

```
1 m1 <- lm("Drinks_per_week ~ Party_Hours_per_week + Gender",
      data=train_data)
2 summary(m1)
3
4 m2 <-  lm("Drinks_per_week ~ Party_Hours_per_week + Gender + Home_Town",
      data=train_data)
5 summary(m2)
```

Таблица 2.1 — Сводная информация по первой модели: Drinks_per_week ~ Party_Hours_per_week + Gender

Coefficient	**Estimate**	**Std. Error**	**t value**	**Pr(>|t|)**
(Intercept)	1.63	0.37	4.39	1.27e-06***
Party_Hours_per_week	1.14	0.03	34.31	<2e-16***
Gender1	-2.25	0.39	-5.73	1.34e-08***

Residuals:

Min: -46.40 1Q: -2.78 Median: -0.511 3Q: 1.19 Max: 43.36

Model Fit:

Residual standard error: 5.98 on 1019 degrees of freedom

Multiple R-squared: 0.54 Adjusted R-squared: 0.55

F-statistic: 614.3 on 2 and 1019 DF p-value: <2.2e-16

Сводная информация по созданным моделям приведена в таб. 2.1 - 2.2. Как можно видеть, добавление преобразованного города не является значимым. Но изменилась константа (Intercept).

Таблица 2.2 — Сводная информация по второй модели: Drinks_per_week ~ Party_Hours_per_week + Gender + Home_Town

Coefficient	**Estimate**	**Std. Error**	**t value**	**Pr(>|t|)**
(Intercept)	1.85	0.45	4.09	4.56e-05***
Party_Hours_per_week	1.15	0.03	34.28	<2e-16***
Gender1	-2.23	0.40	-5.63	2.38e-08***
Home_Town	-0.36	0.41	-0.86	0.39

Residuals:

Min: -46.351 1Q: -2.653 Median: -0.420 3Q: 1.248 Max: 48.499

Model Fit:

Residual standard error: 5.97 on 1018 degrees of freedom

Multiple R-squared: 0.55 Adjusted R-squared: 0.55

F-statistic: 409.7 on 3 and 1018 DF p-value: <2.2e-16

Построим графики обеих моделей (рис. 2.1). Видно, что обе модели дают параллельные графики для каждого пола. Это связано с равенством коэффициента у переменной Party_Hours_per_week.

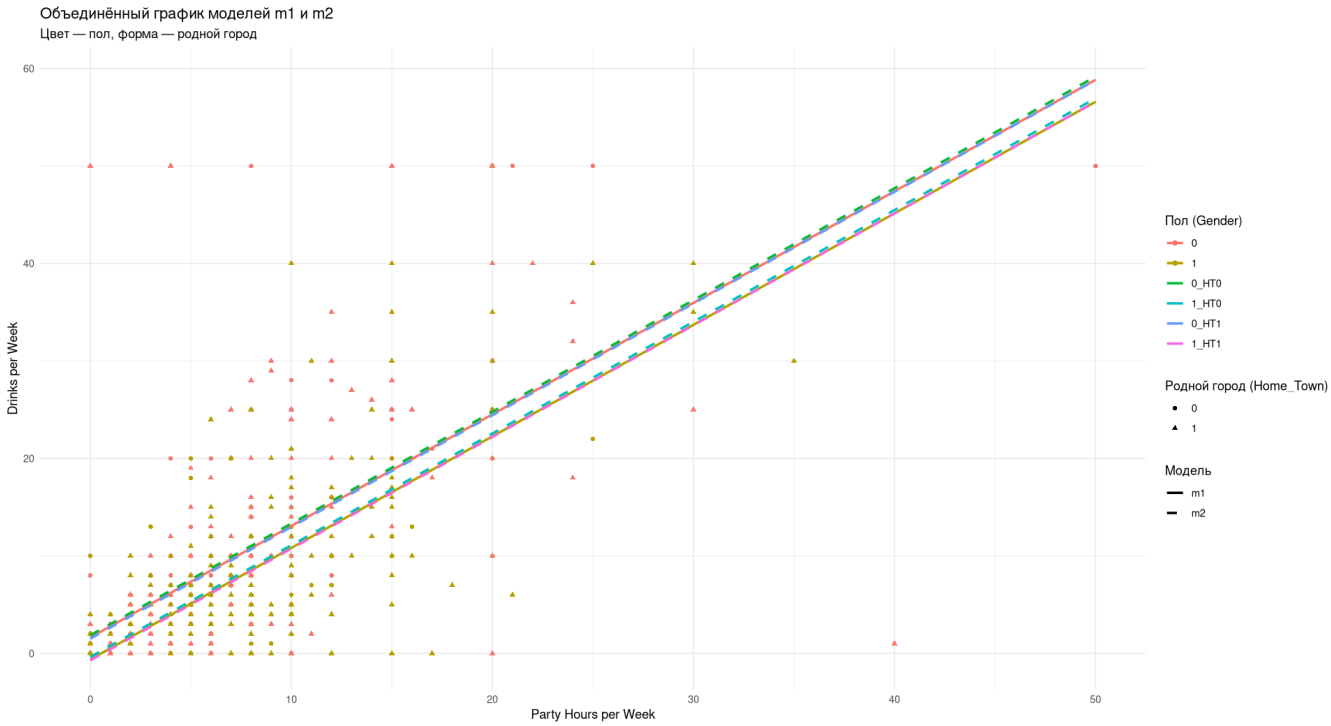

Рисунок 2.1 — Графики моделей

Сделаем предсказание на +5%, +10% и +15% от максимума по переменной вечеринок для обоих полов для первой модели. И создадим доверительный и предиктивный интервалы (рис. 2.2).

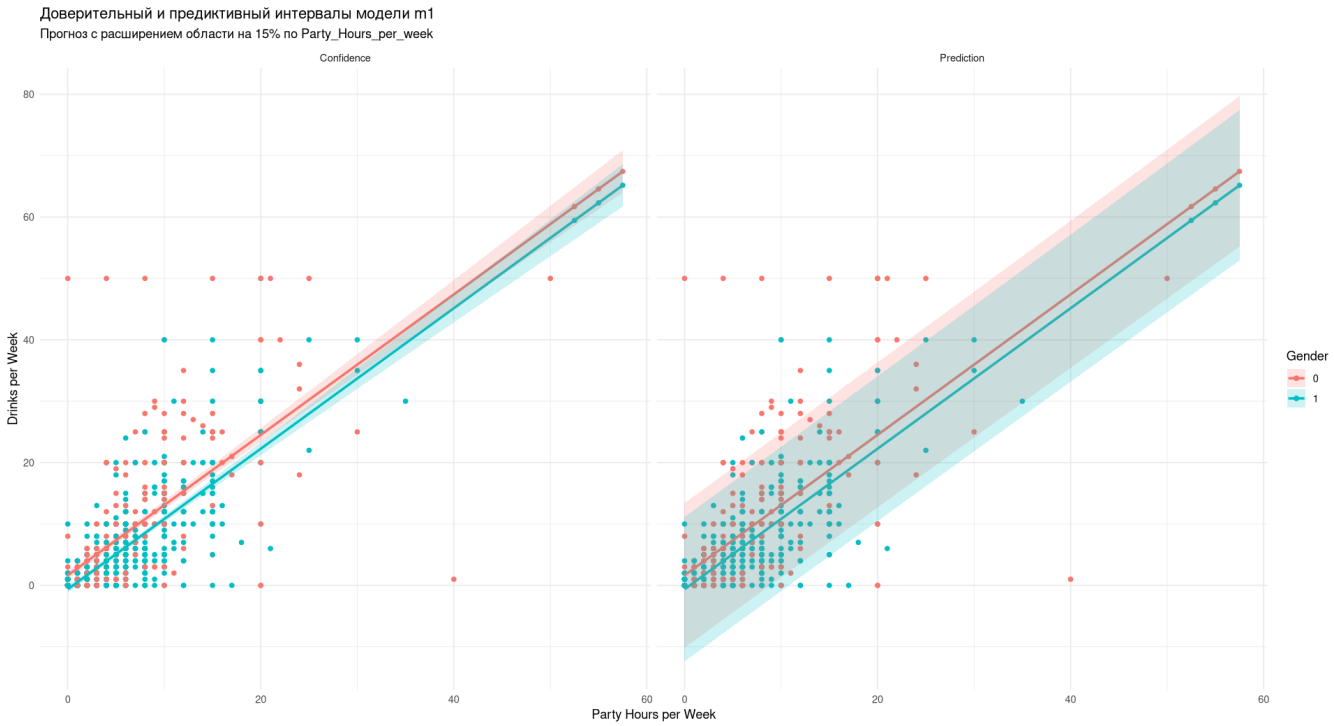

Рисунок 2.2 — Графики интервалов моделей