МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение
высшего образования
«Дальневосточный федеральный университет»
(ДВФУ)

ИНСТИТУТ МАТЕМАТИКИ И КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЙ
Департамент математического и компьютерного моделирования

Курс «Компьютерные методы анализа больших данных»

Лабораторная работа №2
Контрольное мероприятие по рейтингу

на тему «Очистка и сравнительный анализ интернет-данных»
вариант № 3

Выполнил студент Б9122-01.03.02мкт
Вершинин Д.А.
_____

Проверил доцент Достовалов В.Н.

г. Владивосток
2025

Содержание

Введение

Анализ и обработка данных в современном мире занимают центральное место в сфере информационных технологий. Современные методы анализа позволяют выявлять закономерности и зависимости между различными характеристиками в данных, что делает их крайне востребованными в научных и прикладных задачах.

Цель данной работы — продемонстрировать применение методов линейной регрессии на реальных данных. В качестве объекта исследования выбран открытый датасет Stat100_Fall2018_Survey02, предоставленный Университетом Иллинойса. Этот набор данных содержит информацию, собранную в ходе учебного опроса, и хорошо подходит для целей демонстрационного анализа.

В рамках работы будут реализованы две модели линейной регрессии: базовая и расширенная. Сравнение этих моделей позволит выявить, насколько добавление дополнительных признаков влияет на качество предсказания.

Для достижения поставленной цели решаются следующие задачи:

а) Загрузка и предварительная обработка датасета.

б) Построение двух моделей линейной регрессии — базовой и расширенной.

в) Визуализация результатов моделирования для наглядной интерпретации.

г) Сравнительный анализ моделей по ключевым метрикам качества и интерпретация полученных результатов.

В качестве инструмента анализа используется язык R[1]. Для визуализации данных - библиотека ggplot2[2]. Разделение данных произведено с помощью библиотеки caret[3].

1 Подготовка данных

Для начала работы требуется загрузить найденный датасет. Однако заголовки, заданные по умолчанию, содержат поясняющую информацию, поэтому требуется отдельно загрузить данные и их заголовки. Заранее все заголовки были перенесены в файл `headers.txt`. Загрузка основной информации производится 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 заголовков был использован следующий подход (листинг 1.2):

— считывание строки с заголовками;

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

— удаление лишних точек и пробелов;

— разделение значений 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("\\.", "", cleaned_header)
```

```
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, так как последнее значение — NULL).

После проделанных 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 таблице 1.2.

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

№	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 датафрейма (листинг 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)
colSums(is.na(dd))
summary(dd)
```

Пропущеных значение не обнаружено. Размерность полученного датафрейма: (1458, 4).

Описание данных:

а) **Пол (Gender)**:

— 0 = Мужской

— 1 = Женский

б) **Родной город (Home_Town)**: тип населённого 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, проведённых 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 преобразуем пол 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 информация по полученному датафрейму представлена в таблице 1.4.

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

Разделим полученный датафрейм на обучающую и тестовую выборки (70:30) (листинг 1.6).

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, ]
test_data <- dd[-train_index, ]

dim(train_data)
dim(test_data)
```

Размерность тренировочной выборки - (1022, 4).

Тестовой - (436, 4).

Суммарно строк - 1458 (ничего не потеряно).

На этом подготовка данных завершена.

2 Формирование и анализ моделей

2.1 Создание моделей

На основе обучающей выборки, были построены две модели линейной регрессии[4]: базовая (Drinks_per_week от Party_Hours_per_week + Gender) и расширенная (Drinks_per_week от Party_Hours_per_week + Gender + Home_Town). Программный код представлен в листинге 2.1.

Листинг 2.1 — Создание моделей в R

```r
m1 <- lm("Drinks_per_week ~ Party_Hours_per_week + Gender",
    data=train_data)
summary(m1)

m2 <-  lm("Drinks_per_week ~ Party_Hours_per_week + Gender +
    Home_Town", data=train_data)
summary(m2)
```

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

Коэффициент	Оценка	Ст. ошибка	t-стат.	p-значение
(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***

Остатки:

Минимум: -46.40 1 кв.: -2.78 Медиана: -0.511 3 кв.: 1.19

Максимум: 43.36

Качество модели:

Стандартная ошибка остатков: 5.98 при 1019 степенях свободы

Множественный R^2: 0.54 Скорректированный R^2: 0.55

F-статистика: 614.3 при 2 и 1019 СС p-значение: <2.2e-16

Сводная информация по построенным моделям представлена в таблицах 2.1–2.2. Как видно, включение переменной `Home_Town` не является статистически значимым (p = 0.39). Однако изменилось

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

Коэффициент	**Оценка**	**Ст. ошибка**	**t-стат.**	**p-значение**
(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

Остатки:

Минимум: -46.35 1 кв.: -2.65 Медиана: -0.420 3 кв.: 1.25

Максимум: 48.50

Качество модели:

Стандартная ошибка остатков: 5.97 при 1018 степенях свободы

Множественный R^2: 0.55 Скорректированный R^2: 0.55

F-статистика: 409.7 при 3 и 1018 СС p-значение: <2.2e-16

значение свободного члена (Intercept), что указывает на возможное влияние города происхождения на базовый уровень потребления.

2.2 Оценка моделей

Проведём оценку обеих моделей на всей выборке (листинг 2.2).

Листинг 2.2 — Вычисление F-статистики

```
predictions <- predict(m1, newdata = dd)

residuals <- dd$Drinks_per_week - predictions
RSS <- sum(residuals^2)

TSS <- sum((dd$Drinks_per_week - mean(dd$Drinks_per_week))^2)

SSR <- TSS - RSS

k <- length(coef(m1)) - 1
n <- nrow(dd)

F_statistic <- (SSR / k) / (RSS / (n - k - 1))
F_statistic
```

Для первой модели начение F-статистики [4] равно 865.1868. Для второй - 866.3685.

Значения F-статистики намного больше 1, что указывает на то, что обе модели в целом значимы. Вторая модель имеет чуть более высокую F-статистику (866.3685 против 865.1868). Но не стоит говорить о её лучшей объясняющей способности.

На тестовой выборке оценим модели с помощью средне-квадратичного отклонения[4] (листинг 2.3).

Листинг 2.3 — Вычисление MSE

```
predictions_m1 <- predict(m1, newdata=test_data)
predictions_m2 <- predict(m2, newdata=test_data)
mse_m1 <- mean((test_data$Drinks_per_week - predictions_m1)^2)
mse_m2 <- mean((test_data$Drinks_per_week - predictions_m2)^2)

cat("MSE for Model 1 (m1):", mse_m1, "\n")
cat("MSE for Model 2 (m2):", mse_m2)
```

Для первой модели получено значение 41.61. Для второй - 41.58. Вторая модель чуть лучше описывает тестовые данные.

2.3 Графики моделей

Построим графики обеих моделей (рисунок 2.1, код представлен в приложении А).

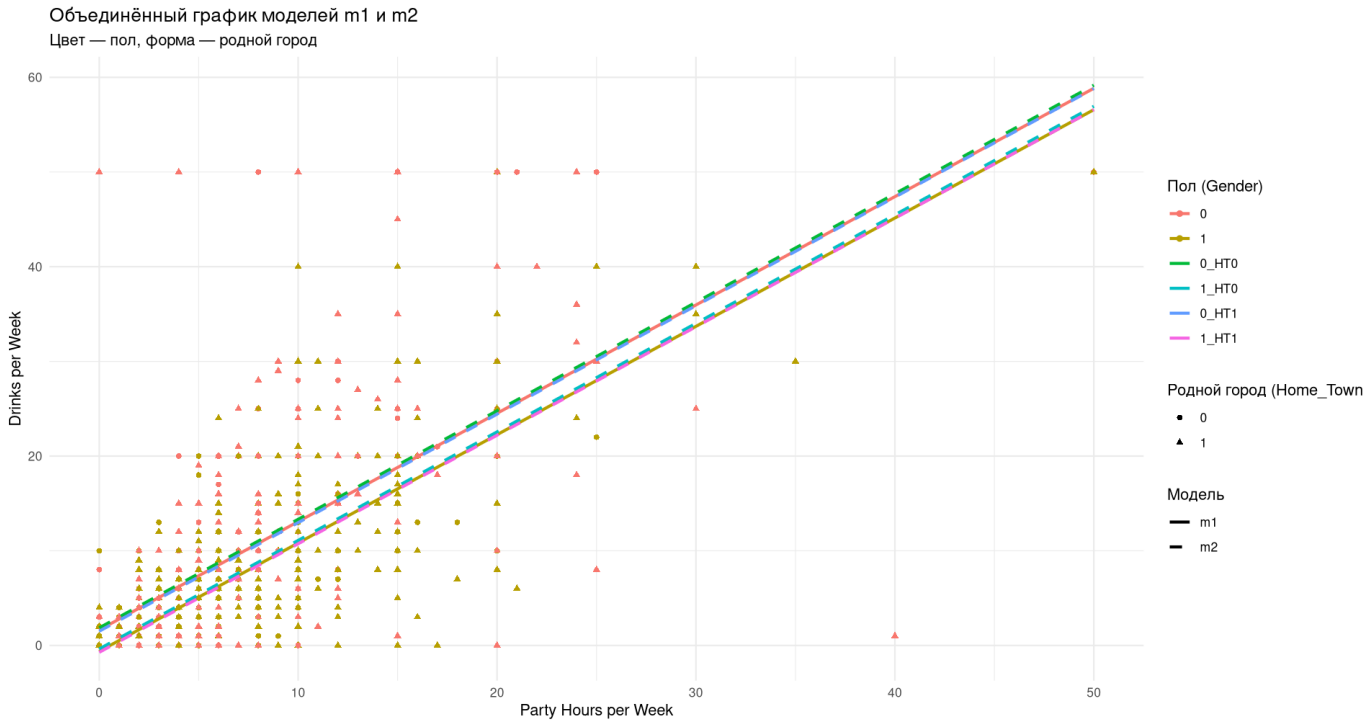

Рисунок 2.1 — Графики регрессионных моделей

На рисунке 2.1 видно, что обе модели дают параллельные прямые для каждого пола. Это объясняется одинаковым коэффициентом переменной `Party_Hours_per_week`, который отражает вклад вечернего поведения вне зависимости от пола.

2.4 Доверительные и предиктивные интервалы

Расчёт новых данных

Рассчитаем предсказания для значений переменной `Party_Hours_per_week`, увеличенных на 5%, 10% и 15% от её максимума, для обоих полов, используя первую модель (таб. 2.3, листинг 2.4).

Листинг 2.4 — Предсказания новых значений

```
x_max <- max(dd$Party_Hours_per_week)
x_ext <- x_max * c(1.05, 1.10, 1.15)

new_rows <- expand.grid(
```

```
  Party_Hours_per_week = x_ext,
  Gender = levels(train_data$Gender)
)

new_rows <- expand.grid(Party_Hours_per_week = x_ext, Gender =
    levels(train_data$Gender))

new_rows$Drinks_per_week <- predict(m1, newdata = new_rows)
new_rows$Home_Town <- NA

conf <- predict(m1, newdata = new_rows, interval = "confidence")
pred <- predict(m1, newdata = new_rows, interval = "prediction")

results <- cbind(new_rows, conf, pred_Lwr = pred[ , "lwr"], pred_Upr =
    pred[ , "upr"])
print(results)
```

Таблица 2.3 — Предсказания и интервалы для переменных

Party_Hours	Gender	Fit	Lwr	Upr	Pred_Lwr	Pred_Upr
52.5	0	61.70	58.58	64.82	49.55	73.86
55.0	0	64.57	61.28	67.85	52.37	76.76
57.5	0	67.43	63.98	70.87	55.19	79.66
52.5	1	59.45	56.32	62.57	47.29	71.60
55.0	1	62.31	59.02	65.59	50.11	74.50
57.5	1	65.17	61.72	68.61	52.93	77.41

Построим доверительные и предиктивные интервалы[4] на всём промежутке (рисунок 2.2, код представлен в приложении Б).

Доверительные интервалы (Confidence)

Доверительные интервалы отображают неопределённость оценки среднего значения зависимой переменной при фиксированном значении независимой переменной. Видно, что интервалы более узкие в центральной части, где плотность наблюдений выше, и расширяются к краям, что типично при экстраполяции.

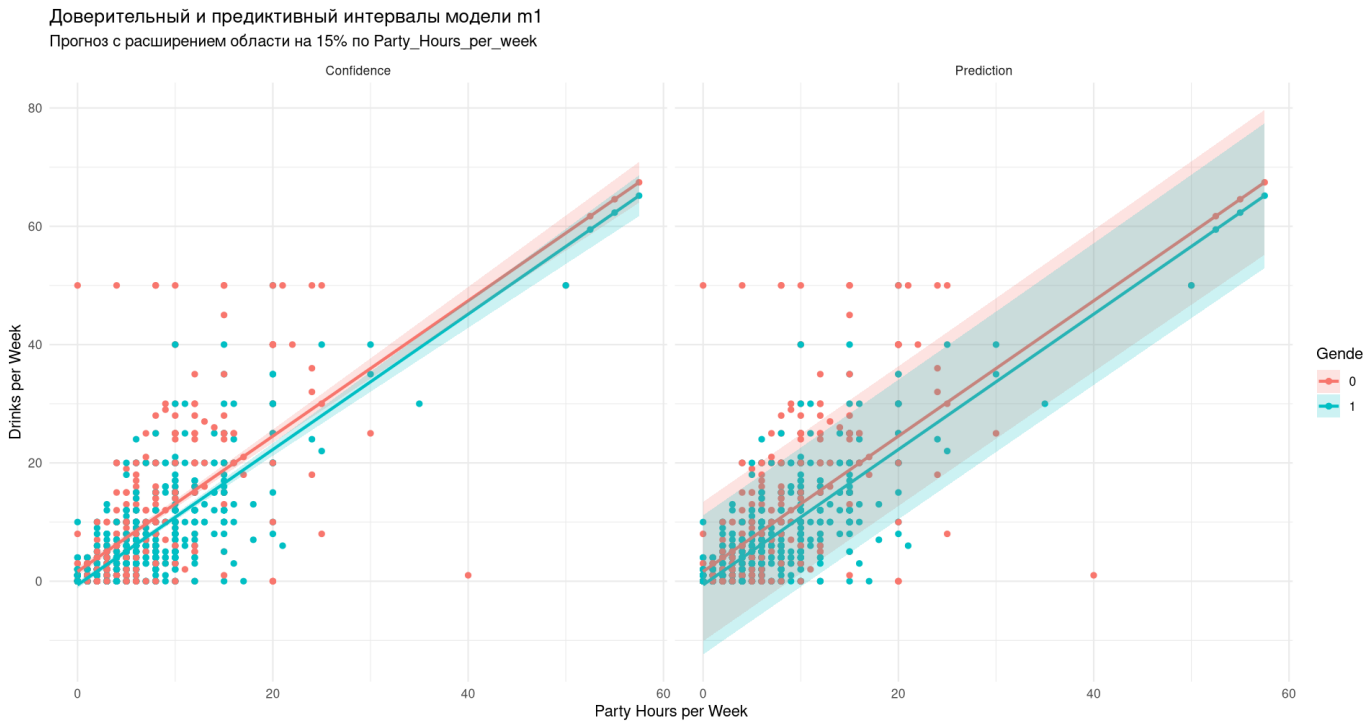

Рисунок 2.2 — Графики доверительных и предиктивных интервалов модели (на фоне точки всей выборки. Справа сверху шесть точек - новые предсказание +5-15% от максимального значения всего датасета)

Предиктивные интервалы (Prediction)

Предиктивные интервалы шире, так как учитывают не только ошибку оценки среднего, но и вариацию индивидуальных значений. Это позволяет:

— Адекватно отображать разброс реальных наблюдений вокруг линии регрессии.

— Учесть существенно большую неопределённость при попытке предсказать индивидуальное поведение.

Область экстраполяции

Интервалы построены с учётом расширения по оси `Party_Hours_per_week` на 15%, что наглядно демонстрирует:

— Резкое расширение обоих типов интервалов за пределами основной области наблюдений.

— Повышенную неопределённость прогнозов вне обучающего диапазона данных.

Выводы

— Модель демонстрирует адекватную уверенность в оценке среднего, но справедливо расширяет предиктивные интервалы, что указывает на реалистичную оценку неопределённости.

— Наблюдаемое различие между полами при сохранении общей тенденции подтверждает обоснованность стратификации по полу.

Заключение

Таким образом, была продемонстрирована практическая реализация методов линейной регрессии. Основная цель исследования была успешно достигнута, что подтверждается выполнением всех поставленных задач.

Был предобработан датасет: преобразован показатели городов и пола, выбраны признаки для анализа.

Построены две модели и проведён анализ с использованием F-статистики и MSE.

Также были созданы и проанализированы графики, которые иллюстрируют поведение моделей.

Список использованных источников

1. *Мэтлофф, Норман.* Искусство программирования на R. Погружение в большие данные / Норман Мэтлофф. Библиотека программиста. — СПб.: Питер, 2019. — С. 416.

2. *Мастицкий, Сергей.* Визуализация данных с помощью ggplot2 / Сергей Мастицкий. — Москва: ДМК Пресс, 2017. — С. 222.

3. *Kuhn, Max.* Caret: Classification and Regression Training. — R package version 6.0-86. — 2019. URL: `https://cran.r-project.org/package=caret/vignettes/caret.html`.

4. *Wasserman, Larry.* All of Statistics: A Concise Course in Statistical Inference / Larry Wasserman. Springer Texts in Statistics. — Springer, 2004.

Приложение А Код графика для регрессионных моделей m1 и m2

Листинг А.1 — Код для формирование графика обеих моделей

```
library(ggplot2)

x_seq <- seq(
  from = min(dd$Party_Hours_per_week),
  to = max(dd$Party_Hours_per_week),
  length.out = 100
)

newdata_m1 <- expand.grid(
  Party_Hours_per_week = x_seq,
  Gender = levels(dd$Gender)
)
newdata_m1$Home_Town <- NA  # Добавляем для совместимости
newdata_m1$Pred <- predict(m1, newdata_m1)
newdata_m1$Model <- "m1"
newdata_m1$Group <- newdata_m1$Gender

newdata_m2 <- expand.grid(
  Party_Hours_per_week = x_seq,
  Gender = levels(dd$Gender),
  Home_Town = c(0, 1)
)
newdata_m2$Pred <- predict(m2, newdata_m2)
newdata_m2$Model <- "m2"
newdata_m2$Group <- interaction(newdata_m2$Gender,
    newdata_m2$Home_Town, sep = "_HT")

pred_all <- rbind(newdata_m1, newdata_m2)
dd$Home_Town <- factor(dd$Home_Town)

ggplot(dd, aes(x = Party_Hours_per_week, y = Drinks_per_week)) +
  geom_point(
    aes(color = Gender, shape = Home_Town),
    alpha = 1
  ) +
  geom_line(
    data = pred_all,
    aes(
      x = Party_Hours_per_week,
      y = Pred,
      color = Group,
```

```
      linetype = Model
    ),
    size = 1
  ) +
  scale_linetype_manual(
    values = c(m1 = "solid", m2 = "dashed"),
    name = "Модель"
  ) +
  labs(
    title = "Объединённый график моделей m1 и m2",
    subtitle = "Цвет - пол, форма - родной город",
    x = "Party Hours per Week",
    y = "Drinks per Week",
    color = "Пол (Gender)",
    shape = "Родной город (Home_Town)"
  ) +
  theme_minimal()
```

Приложение Б Код предсказаний модели m1

Листинг Б.1 — Код для формирование доверительного и предиктивного предсказания модели m1

```
range_x <- range(dd$Party_Hours_per_week)
x_min <- range_x[1]
x_max <- range_x[2]
x_range <- x_max - x_min

x_ext <- seq(
  from = x_min,
  to   = x_max + 0.15 * x_range,
  length.out = 200
)

newdata <- expand.grid(
  Party_Hours_per_week = x_ext,
  Gender = levels(dd$Gender)
)

conf_pred <- predict(m1, newdata = newdata, interval = "confidence",
    level = 0.95)
conf_df <- cbind(newdata, conf_pred)
conf_df$Type <- "Confidence"

pred_pred <- predict(m1, newdata = newdata, interval = "prediction",
    level = 0.95)
pred_df <- cbind(newdata, pred_pred)
pred_df$Type <- "Prediction"

all_preds <- rbind(conf_df, pred_df)

ggplot() +
  geom_point(data = dd, aes(x = Party_Hours_per_week, y =
      Drinks_per_week, color = Gender), alpha = 1) +
  geom_point(data = new_rows, aes(x = Party_Hours_per_week, y =
      Drinks_per_week, color = Gender, ), alpha = 1) +

  geom_line(data = all_preds, aes(x = Party_Hours_per_week, y = fit,
      color = Gender), size = 1) +

  geom_ribbon(
    data = all_preds,
    aes(x = Party_Hours_per_week, ymin = lwr, ymax = upr, fill =
       Gender),
    alpha = 0.2
```

```
37  ) +
38
39  facet_wrap(~Type) +  # Разделение графика: confidence vs prediction
40
41  labs(
42    title = "Доверительный и предиктивный интервалы модели m1",
43    subtitle = "Прогноз с расширением области на 15% по
        Party_Hours_per_week",
44    x = "Party Hours per Week",
45    y = "Drinks per Week"
46  ) +
47  theme_minimal()
```