KEY MACHINE LEARNING CONCEPTS

Explained with Linear Regression

LOADING REQUIRED LIBRARRIES

```
1 library(tidymodels)
2 library(rio)
3 library(kableExtra)
4 library(janitor)
5 DataMockup=import("https://ai.lange-analytics.com/data/DataStudyTimeMockup.rds")
```

WHAT WILL YOU LEARN

- Reviewing the basic idea behind linear regression
- Learning how how to measure predictive quality with Mean Squar Error (MSE).
- Understanding the role of parameters in a machine learning mode in general and in linear regression in particular
- Calculating optimal regression parameters using OLS
- Finding optimal regression parameters by trial and error
- Distinguish between unfitted and fitted models
- Using the `tidymodels` package to split observations from a dataset randomly into a training and testing dataset.
- Understanding how categorical data such as the sex of a person (female/male) can be transformed into numerical dummy variable.
- Being able to distinguish between dummy encoding and one-hot encoding

JUMPING RIGHT INTO IT

UNIVARIATE OLS WITH A REAL WORLD DATASET

Data Description:

- King County House Sale dataset (Kaggle 2015). House sales prices from May 2014 to May 2015 for King County in Washington State.
- Several predictor variables. For now we use only $Sqft$
- We will only use 100 randomly chosen observations from the total of 21,613 observations.
- We only use Sqft as predictor variable for now.

LOADING THE DATA AND ASSIGNING TRAINING AND TESTING DATA (MANUALLY)

- Code

```
   Price Sqft
1 517000 1180
2 236000 1300
3 490000 2800
4 129000 1150
5 257000 1400
6 312500  870
```

HOW MUCH IS A HOUSE WORTH IN KING COUNTY?

A house with average properties should be predicted with an average price!

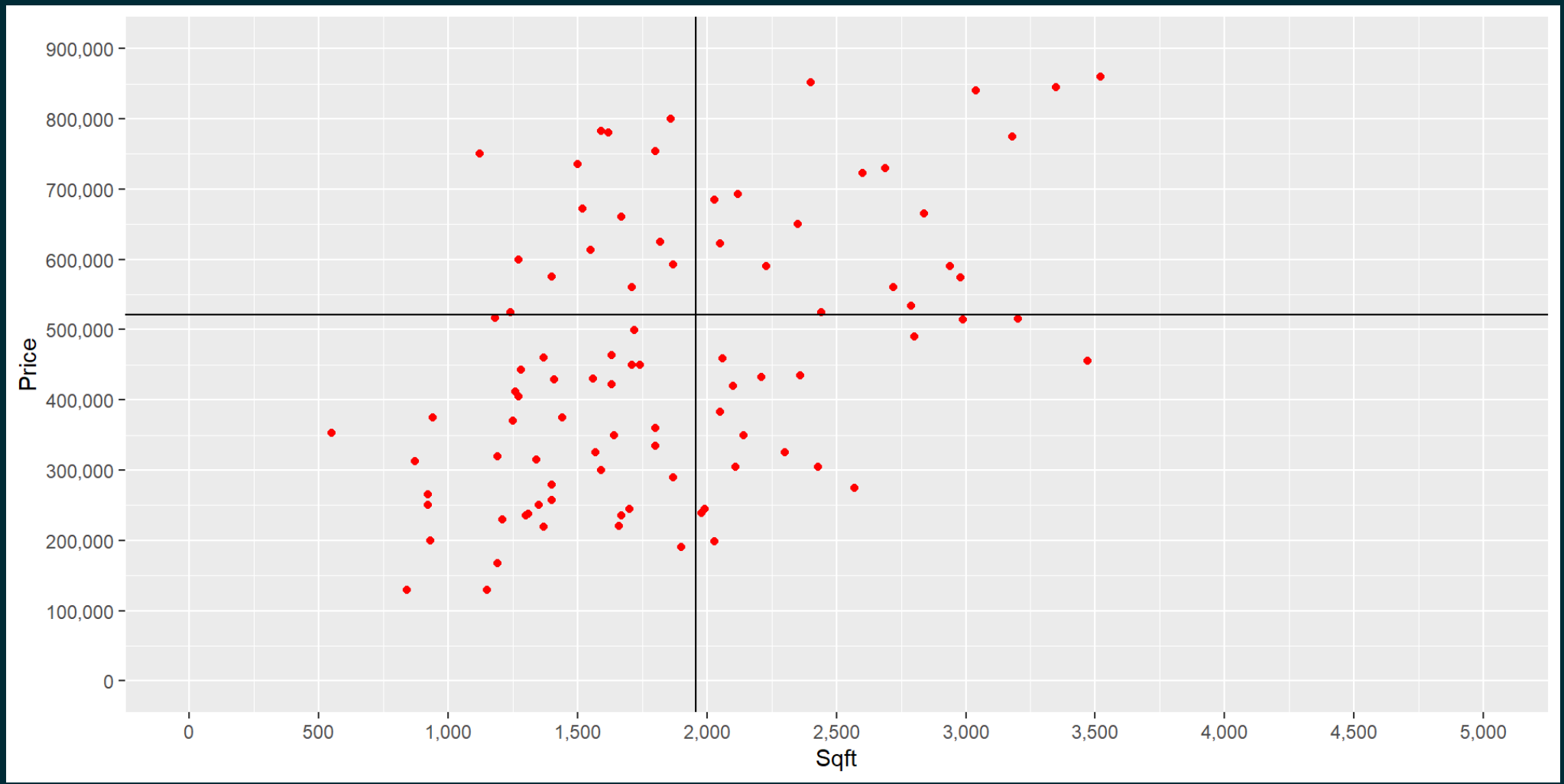
▶ Code

```
The mean square footage of a house in King county is: 1956.7
```

▶ Code

```
The mean price of a house in King county is: 521294.2
```

PREDICTING THE PRICE OF AN AVERAGE SIZED HOUSE AS THE AVERAGE OF ALL HOUSE PRICES



https://ai.lange-analytics.com/

AN INTERACTIVE GRAPH THAT EXPLAINS IT ALL

https://econ.lange-analytics.com/calcat/linregrmeans

FROM UNFITTED TO FITTED MODEL

HOW DOES THE UNFITTED MODEL LOOKS LIKE?

$$\underbrace{\widehat{Price}}_{\hat{y}} = \underbrace{\beta_1}_{m}\underbrace{Sqft}_{x} + \underbrace{\beta_0}_{b}$$

FITTING THEMODEL WITH TIDYMODELS

▶ Code

```
# A tibble: 2 × 5
  term        estimate std.error statistic  p.value
  <chr>          <dbl>     <dbl>     <dbl>    <dbl>
1 (Intercept)   52509.   64183.      0.818 4.15e- 1
2 Sqft            240.      30.6     7.84  5.67e-12
```

UNFITTED MODEL VS FITTED WORKFLOW MODEL

Unfitted Model:

$$\underbrace{\widehat{Price}}_{\hat{y}} = \underbrace{\beta_1}_{m}\ \underbrace{Sqft}_{x} + \underbrace{\beta_0}_{b}$$

▶ Code

```
# A tibble: 2 × 5
  term        estimate std.error statistic  p.value
  <chr>          <dbl>     <dbl>     <dbl>    <dbl>
1 (Intercept)   52509.    64183.      0.818 4.15e- 1
2 Sqft            240.       30.6     7.84  5.67e-12
```

Fitted Model:

$$\underbrace{\widehat{Price}}_{\hat{y}} = \underbrace{240}_{m} \cdot \underbrace{Sqft}_{x} + \underbrace{52509}_{b}$$

INTERPRETATION AND SIGNIFICANCE

```
# A tibble: 2 × 5
  term        estimate std.error statistic  p.value
  <chr>          <dbl>     <dbl>     <dbl>    <dbl>
1 (Intercept)   52509.    64183.     0.818 4.15e- 1
2 Sqft            240.     30.6      7.84  5.67e-12
```

$$\widehat{Price} = 240 \cdot Sqft + 52509$$
$$(+240) = 240 \cdot (+1) + (+0)$$
$$(+480) = 240 \cdot (+2) + (+0)$$
$$(+720) = 240 \cdot (+3) + (+0)$$

For each extra $Sqft$ the predicted price increases by $240

The variable $Sqft$ is significant. I.e., the probability that the related coefficient β_1 equals zero is extremely small.

HOW DOES THE FITTED MODEL THAT CONSIDERS SQFT IMPROVES THE PREDICTION COMPARED TO A SIMPLE AVERAGE

Look at the simulation again and choose 240 for beta 1:

https://econ.lange-analytics.com/calcat/linregrmeans

EVALUATING PREDICTIVE QUALITY WITH THE TESTING DATASET

```
1 DataTestWithPred=augment(WFModelHouses, new_data=DataTest)
2 metrics(DataTestWithPred, truth=Price, estimate=.pred)
```

```
# A tibble: 3 × 3
  .metric .estimator  .estimate
  <chr>   <chr>           <dbl>
1 rmse    standard   163476.
2 rsq     standard        0.626
3 mae     standard   132050.
```

UNIVARIATE LINEAR REGRESSION - DATA TABLE AND GOAL

The Regression:

$$\hat{y}_i = \beta_1 x_i + \beta_2$$

The Goal

Find values for β_1 and β_2 that minimize the prediction errors $(\hat{y}_i - y_i)^2$

The Data Table

Mockup Training Dataset

	y	x
i	Grade	StudyTime
1	65	2
2	82	3
3	93	7
4	93	8
5	83	4

UNIVARIATE LINEAR REGRESSION - DATA DIAGRAM AND GOAL

The Regression:

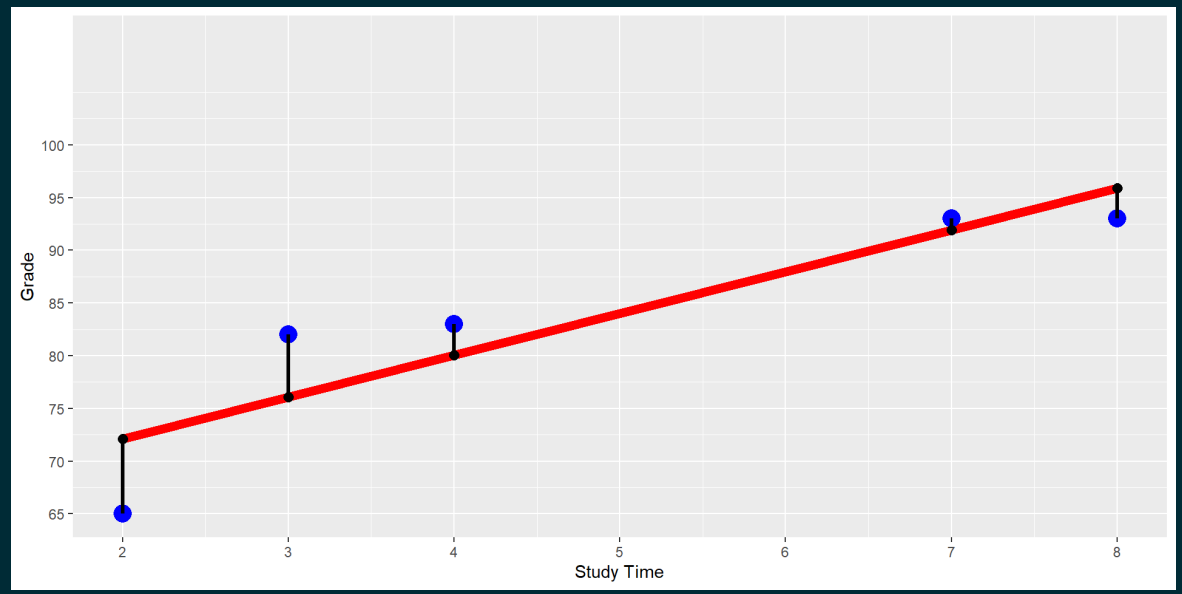
$$\hat{y}_i = \beta_1 x_i + \beta_2$$

The Goal

Find values for β_0 and β_1 that minimize the prediction errors $(\hat{y}_i - y_i)^2$

The Data Diagram

▶ Code



HOW TO MEASURE PREDICTION QUALITY

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2$$

$$\Longleftrightarrow$$

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(\overbrace{\underbrace{\beta_1 x_i + \beta_2}_{\text{Prediction } i} - y_i}^{\text{Error } i})^2$$

Note, when the data are given (i.e., x_i and y_i are given), the MSE depends only on the choice of β_1 and β_2 »

HOW TO MEASURE PREDICTION QUALITY WITH THE MSE

$$MSE = \frac{(\beta_1 x_1 + \beta_2 - y_1)^2 + (\beta_1 x_2 + \beta_2 - y_2)^2 + \cdots + (\beta_1 x_5 + \beta_2 - y_5)^2}{5}$$

$$\Longleftrightarrow$$

$$MSE = \frac{1}{5}\Bigg[(\underbrace{\overbrace{\beta_1 \cdot 2 + \beta_2}^{\text{Prediction 1}} - 65}_{\text{Error 1}})^2 + (\underbrace{\overbrace{\beta_1 \cdot 3 + \beta_2}^{\text{Prediction 2}} - 82}_{\text{Error 2}})^2$$

$$+ (\underbrace{\overbrace{\beta_1 \cdot 7 + \beta_2}^{\text{Prediction 3}} - 93}_{\text{Error 3}})^2 + (\underbrace{\overbrace{\beta_1 \cdot 8 + \beta_2}^{\text{Prediction 4}} - 93}_{\text{Error 4}})^2$$

$$+ (\underbrace{\overbrace{\beta_1 \cdot 4 + \beta_2}^{\text{Prediction 5}} - 83}_{\text{Error 6}})^2 \Bigg]$$

CUSTOM R FUNCTION TO CALCULATE MSE

Function Call:

- Code

```
[1] 29.8
```

Function Definition:»

- Code

HOW TO FIND OPTIMAL VALUES FOR β_1 AND β_2

Method 1:

Calculate optimal values for the parameters (the βs) based on Ordinary Least Squares (OLS) using two formulas (**Note,** this method works only for linear regression)

Method 2:

We can use a **systematic trial and error process**.

METHOD 1: CALCULATE OPTIMAL PARAMETERS (ONLY FOR OLS!)

$$\beta_{1,opt} = \frac{N\sum_{i=1}^{N} y_i x_i - \sum_{i=1}^{N} y_i \sum_{i=1}^{N} x_i}{N\sum_{i=1}^{N} x_i^2 - \left(\sum_{i=1}^{N} x_i\right)^2} = 3.96$$

$$\beta_{2,opt.} = \frac{\sum_{i=1}^{N} y_i - \beta_1 \sum_{i=1}^{N} x_i}{N} = 64.18$$

▶ Code

Mockup Training Dataset

	y	x	y x	x x
i	Grade	StudyTime	GradeXStudyTime	StudyTimeSquared
1	65	2	130	4
2	82	3	246	9
3	93	7	651	49
4	93	8	744	64

	y	x	y x	x x
i	Grade	StudyTime	GradeXStudyTime	StudyTimeSquared
5	83	4	332	16

Column Sums

Grade	StudyTime	GradeXStudyTime	StudyTimeSquared
416	24	2103	142

METHOD 2: USE A SYSTEMATIC TRIAL AND ERROR PROCESS 🤓

- **Grid Search (aka Brute Force):**
  1. For a given range of β_1 and β_2 values, build a table with pairs of all combinations of these βs.
  2. Then use our custom `FctMSE()` command to calculate a MSE for each β pair.
  3. Find the β pair with the lowest MSE
- **Optimizer:** Use the R build-in optimizer. Push the start values for β_1 and β_2 together with the data to the optimizer as arguments. The rest is done by the optimizer.
- See the R script in the footnote to see both algorithms in action.»

MULTIVARIATE OLS WITH A REAL WORLD DATASET

MULTIVARIATE OLS WITH A REAL WORLD DATASET

Data

▶ Code

- King County House Sale dataset (Kaggle 2015). House sales prices from May 2014 to May 2015 for King County in Washington State.
- Several predictor variables.
- We will use all 21,613 observations.

MULTIVARIATE ANALYSIS – THREE PREDICTOR VARIABLES

`Sqft`: Living square footage of the house

`Grade` Indicates the condition of houses (1 (worst) to 13 (best))

`Waterfront`: Is house located at the waterfront (`yes` or `no`)

- Code

Unfitted Model: »

$$Price = \beta_1 Sqft + \beta_2 Grade + \beta_3 Waterfront_{yes} + \beta_4$$

MULTIVARIATE REAL WORLD DATASET – SPLITTING

▶ Code

DataTrain

	Price	Sqft	Grade	Waterfront
1	221900	1180	7	no
2	180000	770	6	no
3	189000	1200	7	no
4	230000	1250	7	no
5	252700	1070	7	no
6	240000	1220	7	no

DataTest

	Price	Sqft	Grade	Waterfront
1	1230000	5420	11	no
2	257500	1715	7	no
3	291850	1060	7	no
4	229500	1780	7	no
5	530000	1810	7	no
6	650000	2950	9	no

DUMMY AND ONE-HOT ENCODING

One-Hot Encoding

▶ Code

```
# A tibble: 4 × 2
  Waterfront_yes Waterfront_no
           <dbl>         <dbl>
1              0             1
2              0             1
3              1             0
4              0             1
```

One-hot encoding is easier to interpret but causes problems in OLS (dummy trap) because one variable is redundant. We can calculate one variable from the other (*perfect multicollinearity*):

$$Waterfront_{yes} = 1 - Waterfront_{no}$$

DUMMY AND ONE-HOT ENCODING

Dummy Coding

We use one variable less than we have categories. Waterfront has two categories. Therefore, we use one variable (e.g., `Waterfront_yes`):

Dummy Encoding Example

▶ Code

```
# A tibble: 4 × 1
  Waterfront_yes
           <dbl>
1              0
2              0
3              1
4              0
```

Note, dummy encoding can be done with `step_dummy()` in a *tidymodels recipe*.»

MULTIVARIATE ANALYSIS – BUILDING THE RECIPE

```
1  RecipeHouses=recipe(Price ~ ., data=DataTrain) |>
2                      step_dummy(Waterfront)
```

Here is how the recipe later on (in the workflow) transforms the data:

- Code

```
# A tibble: 6 × 4
   Sqft Grade  Price Waterfront_yes
  <int> <int>  <dbl>          <dbl>
1  1180     7 221900              0
2   770     6 180000              0
3  1200     7 189000              0
4  1250     7 230000              0
5  1070     7 252700              0
6  1220     7 240000              0
```

MULTIVARIATE ANALYSIS – BUILDING THE MODEL DESIGN

Unfitted Model:

```
1 ModelDesignHouses=linear_reg() |>
2   set_engine("lm") |>
3   set_mode("regression")
4 print(ModelDesignHouses)
```

```
Linear Regression Model Specification (regression)

Computational engine: lm
```

»

MULTIVARIATE ANALYSIS – CREATING WORKFLOW & FITTING TO THE TRAINING DATA

- Code

```
# A tibble: 4 × 5
  term           estimate std.error statistic   p.value
  <chr>             <dbl>     <dbl>     <dbl>     <dbl>
1 (Intercept)    -570056.  15133.       -37.7 6.63e-297
2 Sqft              180.      3.25       55.2 0
3 Grade           95214.   2548.         37.4 1.65e-292
4 Waterfront_yes 868338.  22200.         39.1 7.12e-319
```

- Code

```
# A tibble: 1 × 12
  r.squared adj.r.squared   sigma statistic p.value    df   logLik    AIC    BIC
      <dbl>         <dbl>   <dbl>     <dbl>   <dbl> <dbl>    <dbl>  <dbl>  <dbl>
1     0.581         0.581 238574.     7002.       0     3 -208785. 4.18e5 4.18e5
# ℹ 3 more variables: deviance <dbl>, df.residual <int>, nobs <int>
```

MULTIVARIATE ANALYSIS – PREDICTING TESTING DATA AND METRICS

▶ Code

```
# A tibble: 3 × 3
  .metric .estimator  .estimate
  <chr>   <chr>           <dbl>
1 rmse    standard   244656.
2 rsq     standard        0.549
3 mae     standard   163358.
```

EXERCISE

Run the Analysis»

https://ai.lange-analytics.com/exc/?file=05-LinRegrExerc100.Rmd