

Insert Instructor Name

Title, Company

LEARNING OBJECTIVES

- ▶ Define data modeling and simple linear regression
- ▶ Build a linear regression model using a dataset that meets the linearity assumption using the sci-kit learn library
- ▶ Understand and identify multicollinearity in a multiple regression.

PRE-WORK

PRE-WORK REVIEW

- ▶Effectively show correlations between an independent variable x and a dependent variable y
- ▶Be familiar with the get_dummies function in pandas
- ▶ Understand the concepts of outliers and distance.
- ▶ Be able to interpret p values and confidence intervals

OPENING

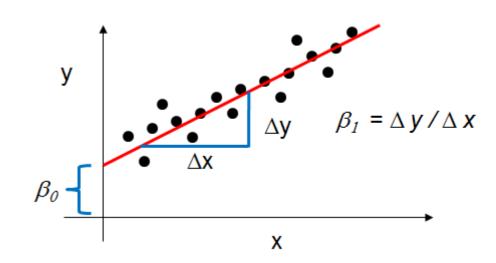
INTRODUCTION TO REGRESSION ANALYSIS

WHERE ARE WE IN THE DATA SCIENCE WORKFLOW?

- Data has been **acquired** and **parsed**.
- ▶ Today we'll **refine** the data and **build** models.
- ▶ We'll also use plots to **represent** the results.

INTRODUCTION

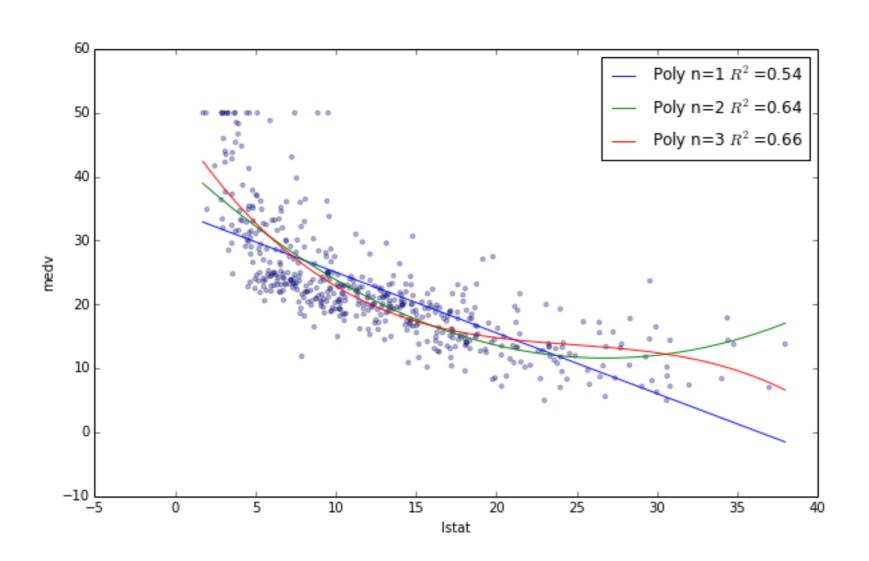
- ▶ Def: Explanation of a continuous variable given a series of independent variables
- The simplest version is just a line of best fit: y = mx + b
- ▶ Explain the relationship between **x** and **y** the starting point **b** and the power in



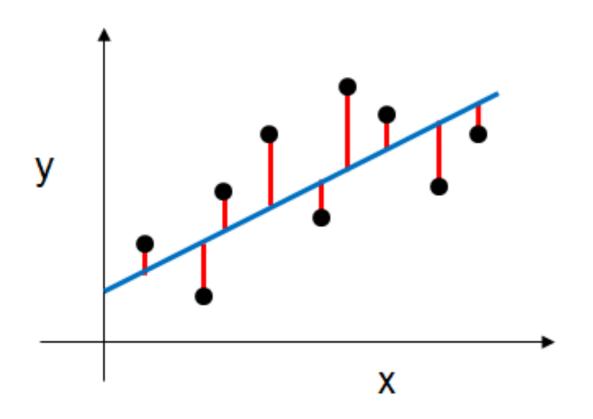
- ▶ However, linear regression uses linear algebra to explain the relationship between *multiple* x's and y.
- ▶ The more sophisticated version: y = beta * X + alpha (+ error)
- Explain the relationship between the matrix **X** and a dependent vector **y** using a y-intercept **alpha** and the relative coefficients **beta**.

R-SQUARED AND RESIDUALS

WHAT IS A RESIDUAL?



LEAST SQUARES



$$SS_{residuals} = \sum_{i=1}^{N} (\hat{y}_i - y_i)^2$$
 Observed Result

WHAT IS R-SQUARED?

- ▶ R-squared, the central metric introduced for linear regression
- ▶ R-squared measures what % of the total variation in *y* is described by the variation in *x*.
- ▶ Which model performed better, one with an r-squared of 0.79 or 0.81?
- ▶ We'll explore loss functions and find ways to refine our model.

Checks

SIMPLE LINEAR REGRESSION ASSUMPTIONS

- Linear regression works **best** when:
 - The data is normally distributed (but doesn't have to be)
 - X's significantly explain y (have low p-values)
 - X's are independent of each other (low multicollinearity)
 - ▶ Resulting values pass linear assumption (depends upon problem)
- If data is not normally distributed, we could introduce bias.

REGRESSING AND NORMAL DISTRIBUTIONS

LAB: REGRESSING AND NORMAL DISTRIBUTIONS

- ▶ Follow along with your starter code notebook while I walk through these examples.
- ▶ The first plot shows a relationship between two values, though not a linear solution.
- ▶ Note that Implot() returns a straight line plot.
- ▶ However, we can transform the data, both log-log distributions to get a linear solution.

GUIDED PRACTICE

USING SEABORN TO GENERATE SIMPLE LINEAR MODEL PLOTS

ACTIVITY: GENERATE SINGLE VARIABLE LINEAR MODEL PLOTS

DIRECTIONS (15 minutes)



1. Update and complete the code in the starter notebook to use **Implot** and display correlations between body weight and two dependent variables: **sleep_rem** and **awake**.

DELIVERABLE

Two plots

SIGNIFICANCE IS KEY

DEMO: SIGNIFICANCE IS KEY

- ▶ Follow along with your starter code notebook while I walk through these examples.
- ▶ What does the residual plot tell us?
- ▶ How can we use the linear assumption?

GUIDED PRACTICE

USING THE LINEAR REGRESSION OBJECT

ACTIVITY: USING THE LINEAR REGRESSION OBJECT

DIRECTIONS (15 minutes)



- 1. With a partner, generate two more models using the log-transformed data to see how this transform changes the model's performance.
- 2. Use the code on the following slide to complete #1.

DELIVERABLE

Two new models

ACTIVITY: USING THE LINEAR REGRESSION OBJECT

DIRECTIONS (15 minutes)

Two new models



DELIVERABLE

Two new models

INTRODUCTION

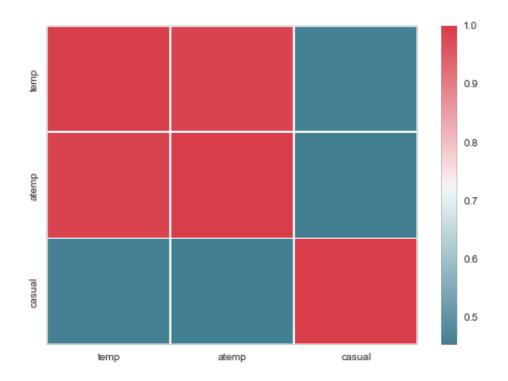
MULTIPLE REGRESSION ANALYSIS

MULTIPLE REGRESSION ANALYSIS

- Simple linear regression with one variable can explain some variance, but using multiple variables can be much more powerful.
- ▶ We want our multiple variables to be mostly independent to avoid multicollinearity.
- ▶ Multicollinearity, when two or more variables in a regression are highly correlated, can cause problems with the model.

BIKE DATA EXAMPLE

- We can look at a correlation matrix of our bike data.
- ▶ Even if adding correlated variables to the model improves overall variance, it can introduce problems when explaining the output of your model.
- ▶ What happens if we use a second variable that isn't highly correlated with temperature?



GUIDED PRACTICE

MULTICOLLINEARITY WITH DUMMY VARIABLES

ACTIVITY: MULTICOLLINEARITY WITH DUMMY VARIABLES

DIRECTIONS (15 minutes)



- 1. Load the bike data.
- 2. Run through the code on the following slide.
- 3. What happens to the coefficients when you include all weather situations instead of just including all except one?

DELIVERABLE

Two models' output

ACTIVITY: MULTICOLLINEARITY WITH DUMMY VARIABLES

DIRECTIONS (15 minutes)



```
lm = linear_model.LinearRegression()
weather = pd.get_dummies(bike_data.weathersit)
get_linear_model_metrics(weather[[1, 2, 3, 4]], y, lm)
print
# drop the least significant, weather situation = 4
get_linear_model_metrics(weather[[1, 2, 3]], y, lm)
```

DELIVERABLE

Two models' output

GUIDED PRACTICE

COMBINING FEATURES INTO A BETTER MODEL

ACTIVITY: COMBINING FEATURES INTO A BETTER MODEL

DIRECTIONS (15 minutes)



- 1. With a partner, complete the code on the following slide.
- 2. Visualize the correlations of all the numerical features built into the dataset.
- 3. Add the three significant weather situations into our current model.
- 4. Find two more features that are not correlated with the current features, but could be strong indicators for predicting guest riders.

DELIVERABLE

Visualization of correlations, new models

ACTIVITY: COMBINING FEATURES INTO A BETTER MODEL



DIRECTIONS (15 minutes)

```
lm = linear model.LinearRegression()
bikemodel data = bike data.join() # add in the three weather
situations
cmap = sns.diverging palette(220, 10, as cmap=True)
correlations = # what are we getting the correlations of?
print correlations
print sns.heatmap(correlations, cmap=cmap)
columns to keep = [] #[which variables?]
final feature set = bikemodel data[columns to keep]
get_linear_model_metrics(final_feature_set, y, lm)
DELIVERABLE
Visualization of correlations, new models
```

INDEPENDENT PRACTICE

BUILDING MODELS FOR OTHER Y VARIABLES

ACTIVITY: BUILDING MODELS FOR OTHER Y VARIABLES



DIRECTIONS (25 minutes)

- Build a new model using a new y variable: registered riders.
- 2. Pay attention to the following:
 - a. the distribution of riders (should we rescale the data?)
 - b. checking correlations between the variables and y variable
 - c. choosing features to avoid multicollinearity
 - d. model complexity vs. explanation of variance
 - e. the linear assumption

BONUS

- 1. Which variables make sense to dummy?
- 2. What features might explain ridership but aren't included? Can you build these features with the included data and pandas?

DELIVERABLE

A new model and evaluation metrics

SIMPLE REGRESSION ANALYSIS IN SKLEARN

SIMPLE LINEAR REGRESSION ANALYSIS IN SKLEARN

- ▶ Sklearn defines models as *objects* (in the OOP sense).
- ▶ You can use the following principles:
 - All sklearn modeling classes are based on the <u>base estimator</u>. This means all models take a similar form.
 - ightharpoonup All estimators take a matrix \mathbf{X} , either sparse or dense.
 - ▶Supervised estimators also take a vector **y** (the response).
 - Estimators can be customized through setting the appropriate parameters.

CLASSES AND OBJECTS IN OBJECT ORIENTED PROGRAMMING

- ▶ **Classes** are an abstraction for a complex set of ideas, e.g. *human*.
- > Specific **instances** of classes can be created as **objects**.

```
▶john_smith = human()
```

▶ Objects have **properties**. These are attributes or other information.

```
>john_smith.age
>john_smith.gender
```

Diject have **methods**. These are procedures associated with a class/object.

```
>john_smith.breathe()
>john_smith.walk()
```

SIMPLE LINEAR REGRESSION ANALYSIS IN SKLEARN

• General format for sklearn model classes and methods

```
# generate an instance of an estimator class
estimator = base_models.AnySKLearnObject()
# fit your data
estimator.fit(X, y)
# score it with the default scoring method (recommended to use the metrics module in the future)
estimator.score(X, y)
# predict a new set of data
estimator.predict(new_X)
# transform a new X if changes were made to the original X while fitting
estimator.transform(new_X)
```

- LinearRegression() doesn't have a transform function
- ▶ With this information, we can build a simple process for linear regression.

CONCLUSION

TOPIC REVIEW

CONCLUSION

- You should now be able to answer the following questions:
 - ▶ What is simple linear regression?
 - ▶What makes multi-variable regressions more useful?
 - ▶What challenges do they introduce?
 - ▶ How do you dummy a category variable?
 - ▶ How do you avoid a singular matrix?

UPCOMING WORK

UPCOMING WORK

Week 4: Lesson 8

▶ Project: Final Project, Deliverable 1

Q & A

EXIT TICKET

DON'T FORGET TO FILL OUT YOUR EXIT TICKET!