GENERAL ASSEMBLY

INTRODUCTION TO REGRESSION ANALYSIS

Angelo Klin
Katra Analytics

LEARNING OBJECTIVES

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

PRE-WORK

PRE-WORK REVIEW

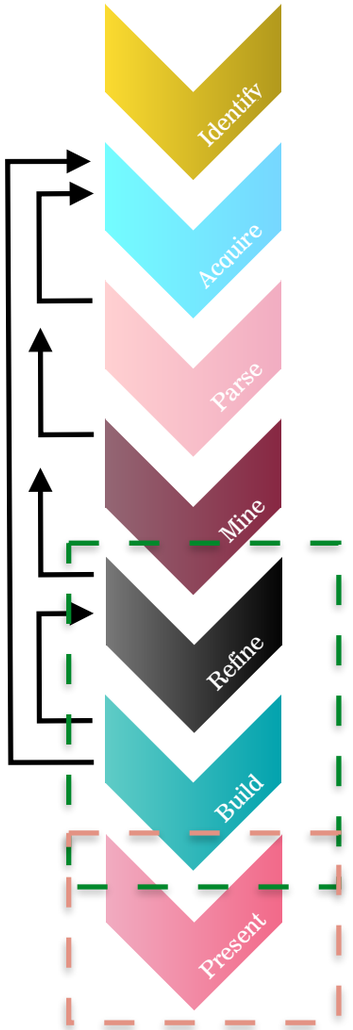
- Effectively show correlations between an Independent Variable **X** and a Dependent Variable **Y**
- Be familiar with the `get_dummies()` function in Pandas
- Understand the difference between vectors, matrices, Series and DataFrames
- Understand the concepts of outliers and distance
- Be able to interpret 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 will Refine the Data and Build Models
- We will also use plots to Present the results



DATA SCIENCE PROBLEMS

- Supervised
  - Both inputs and outcome are in the data

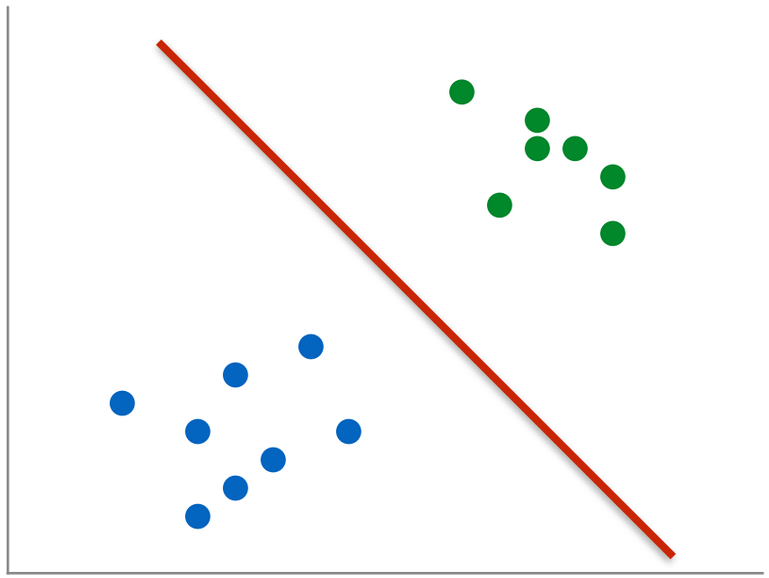
- Unsupervised
  - Only inputs are in the data

REGRESSION / CLASSIFICATION

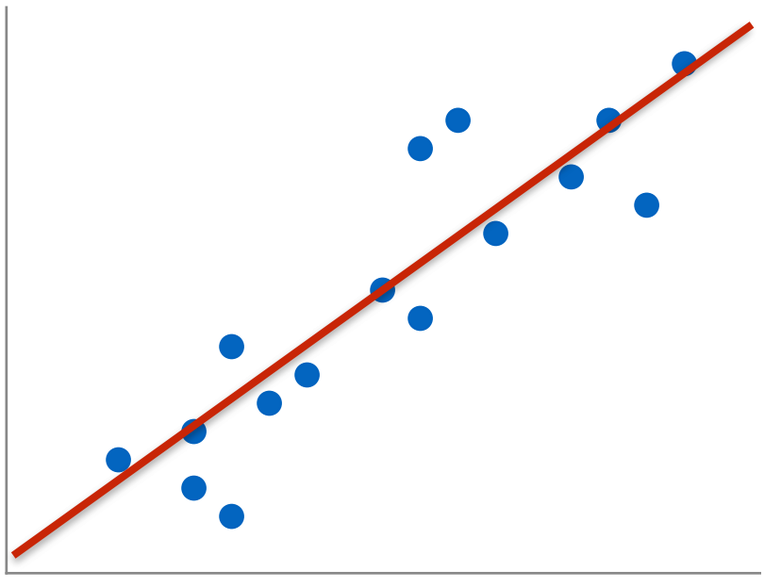
- Regression results can have a value range from -∞ to ∞

- Classification is used when predicted values (i.e. class labels) are not greater than or less than each other

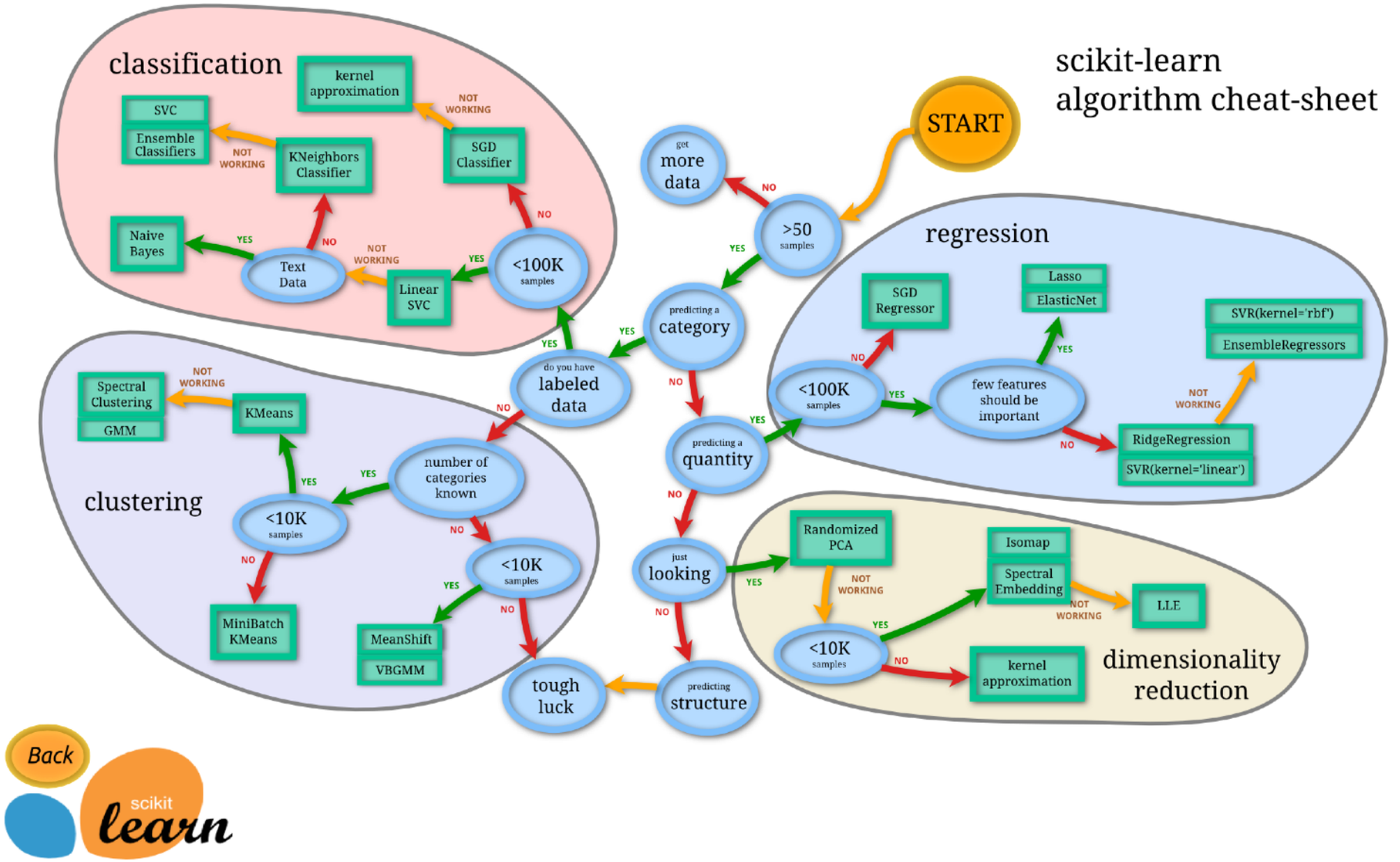
Classification



Regression

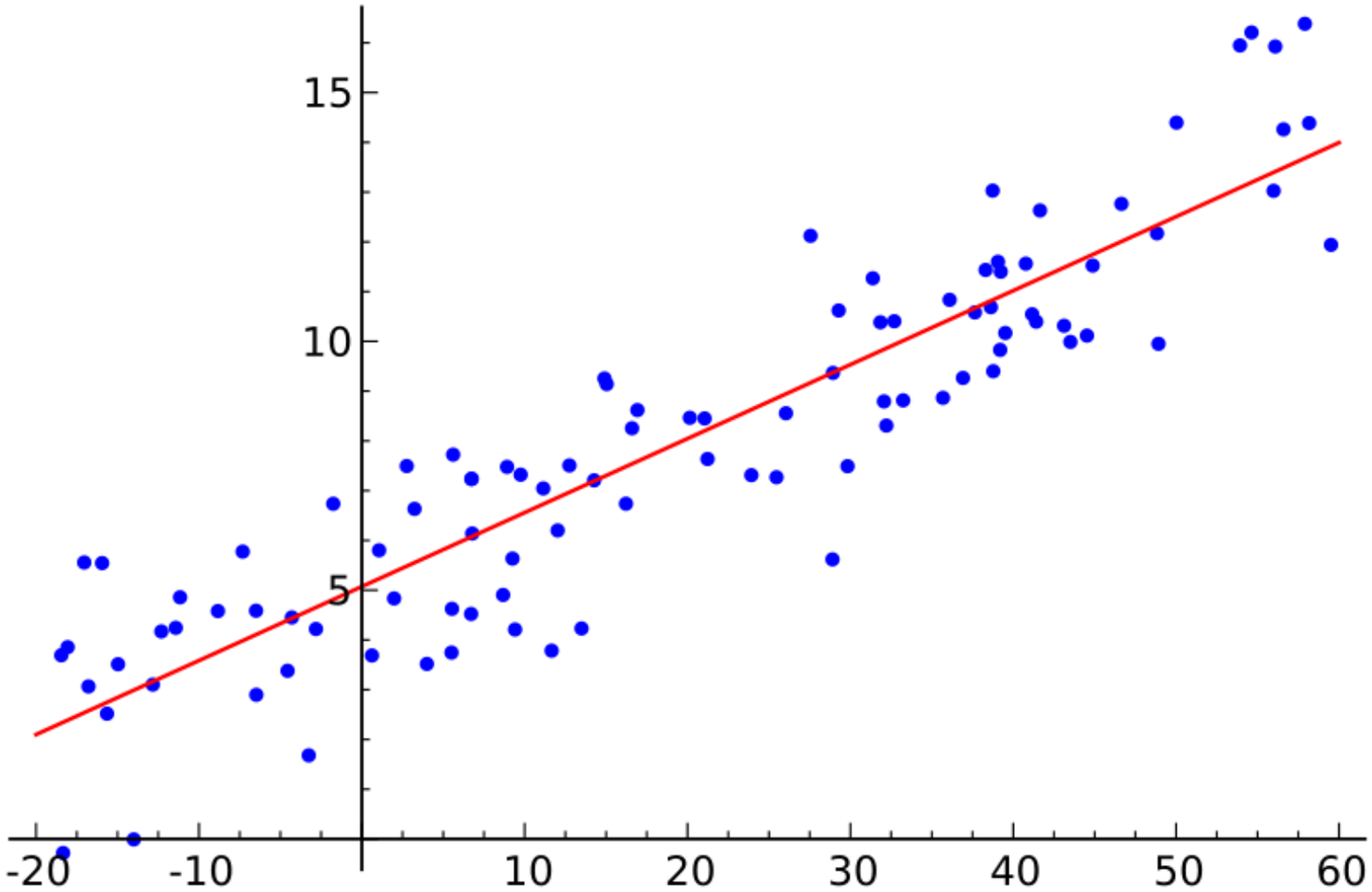


DATA SCIENCE PROBLEMS



SIMPLE LINEAR REGRESSION

SIMPLE LINEAR REGRESSION

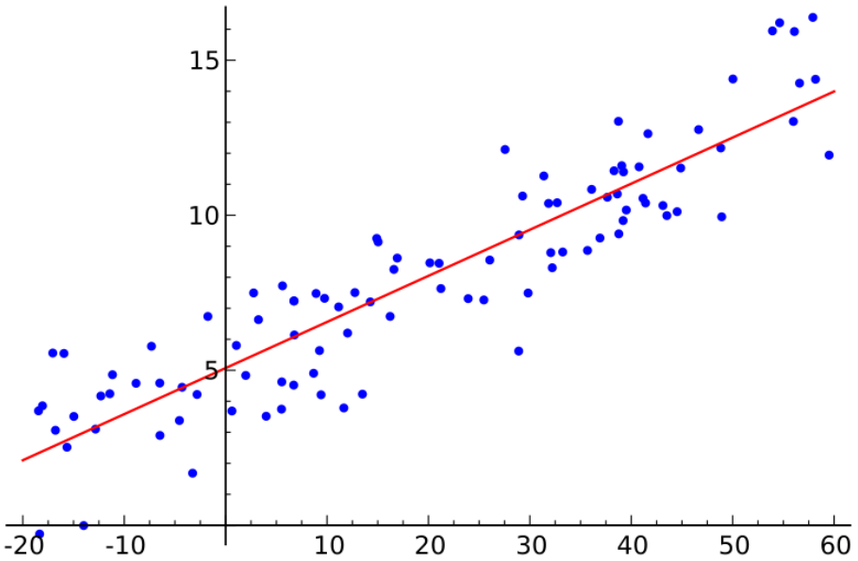


SIMPLE LINEAR REGRESSION

- Explanation of a continuous variable given a series of independent variables

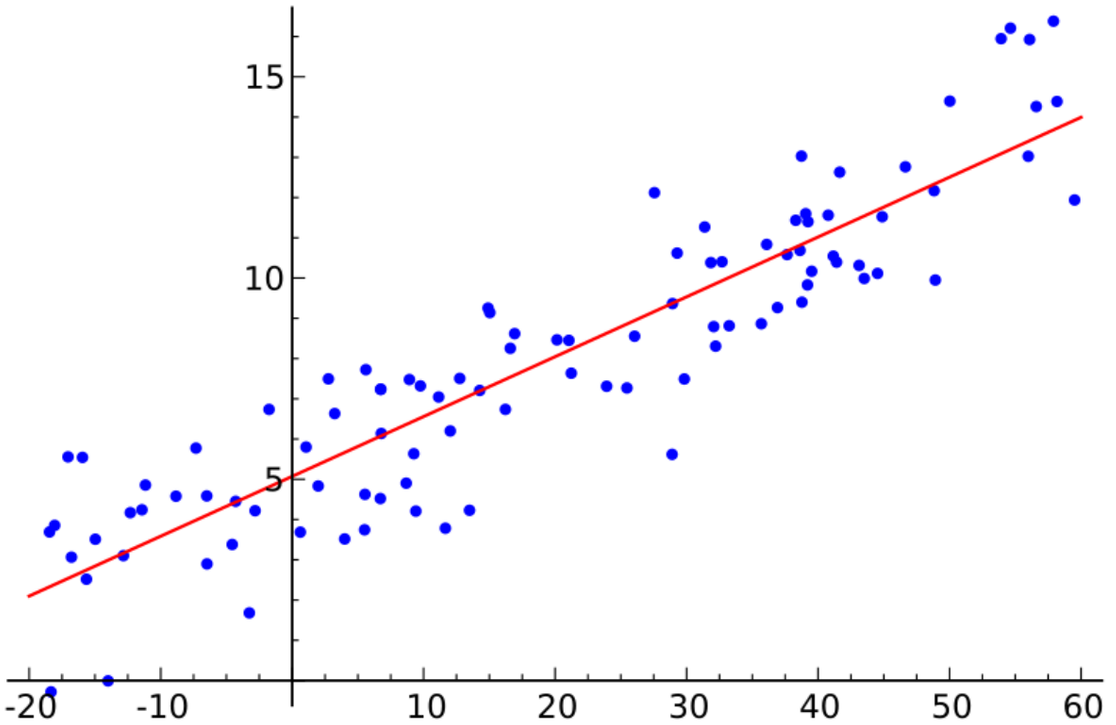
- Explain the relationship between x and y using the starting point a and the power in explanation b

- The simplest version is just a line of best fit $y = a + bx$



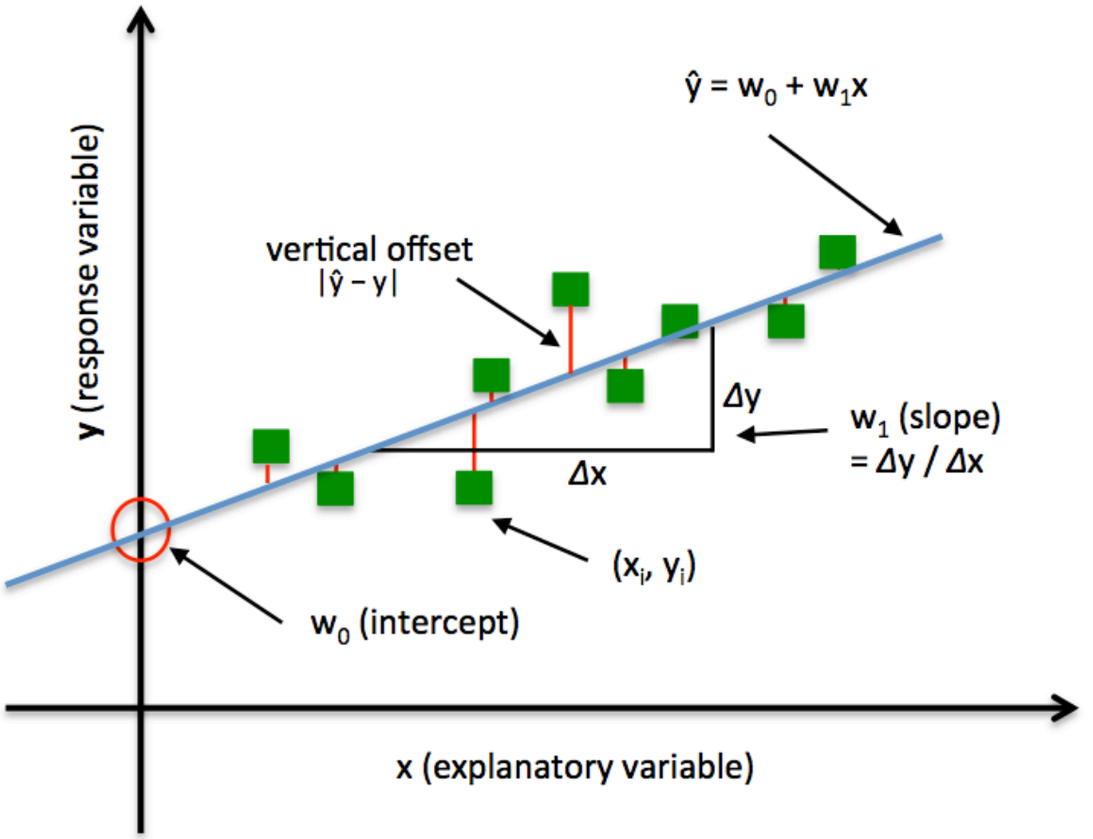
SIMPLE LINEAR REGRESSION

$$y = a + bx$$



SIMPLE LINEAR REGRESSION

$$y = a + bx \equiv w_0 + w_1 x$$



LINEAR REGRESSION

- However, linear regression uses linear algebra to explain the relationship between y and **multiple** x

$$y = \alpha + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_m x_m + \epsilon$$

- The more condensed Matrix version

$$y = \alpha + X \cdot \beta + \epsilon$$

- Explains the relationship between the
  - Matrix X and a Dependent Vector y
  - Using a y-intercept *alpha* and the relative coefficients *beta*

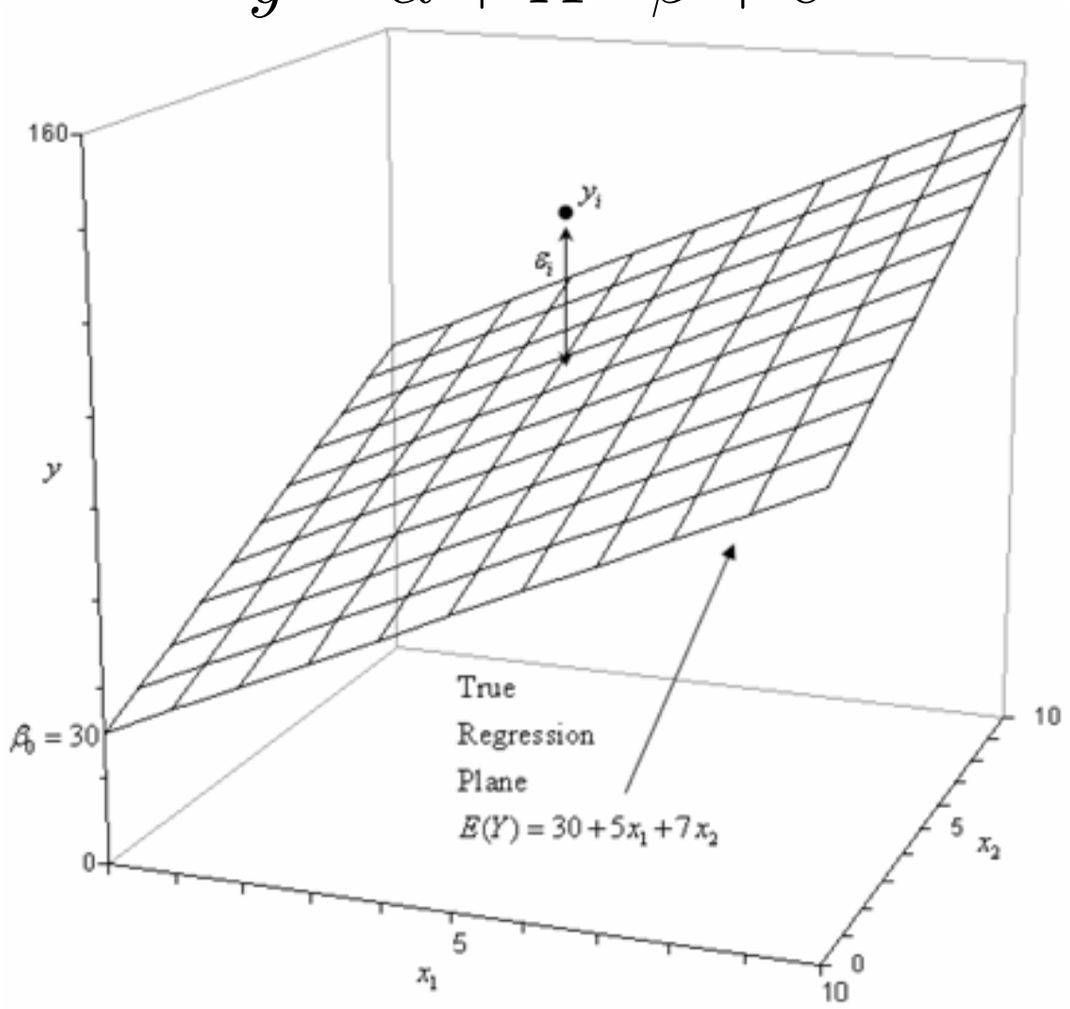
LINEAR REGRESSION

$$y = \alpha + X \cdot \beta + \epsilon$$

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \dots \\ y_n \end{bmatrix} = \begin{bmatrix} 1 & x_{1,1} & x_{1,2} & x_{1,3} & \cdots & x_{1,m} \\ 1 & x_{2,1} & x_{2,2} & x_{2,3} & \cdots & x_{2,m} \\ 1 & x_{3,1} & x_{3,2} & x_{3,3} & \cdots & x_{3,m} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 1 & x_{n,1} & x_{n,2} & x_{n,3} & \cdots & x_{n,m} \end{bmatrix} \cdot \begin{bmatrix} \alpha \\ \beta_1 \\ \beta_2 \\ \beta_3 \\ \dots \\ \beta_m \end{bmatrix} + \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \epsilon_3 \\ \dots \\ \epsilon_n \end{bmatrix}$$

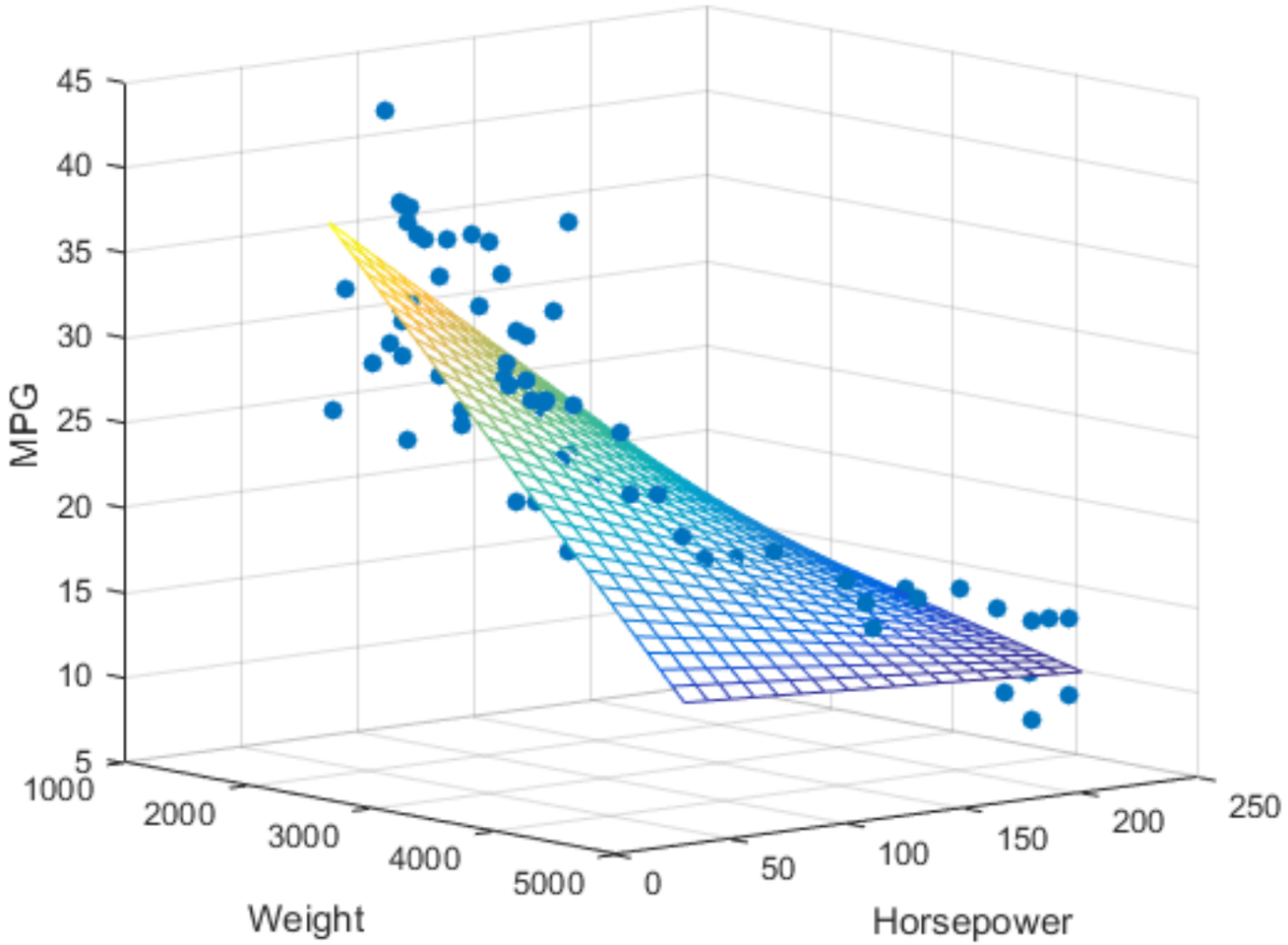
LINEAR REGRESSION

$$y = \alpha + X \cdot \beta + \epsilon$$



LINEAR REGRESSION

$$y = \alpha + X \cdot \beta + \epsilon \therefore MPG = \alpha + \beta_1 Weight + \beta_2 Horsepower + \epsilon$$



LINEAR REGRESSION

- Linear regression works best when
  - The data is normally distributed (but does not have to be)
  - Xs significantly explain y (have low p-values)
  - Xs 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

DEMONSTRATION

REGRESSION AND NORMAL DISTRIBUTIONS

DEMONSTRATION: REGRESSION AND NORMAL DISTRIBUTIONS

- Follow along with your code
  - ~/lessons/lesson-06/code/Linear Regression with Statsmodels and Scikit-Learn.ipynb
- The first plot shows a relationship between two values, though not a linear solution
- Note that lmplot() 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 lmplot() and display correlations between body weight and two dependent variables: sleep_rem and awake

   a. ~/lessons/lesson-06/code/starter/starter-6.ipynb

SIMPLE REGRESSION ANALYSIS IN SCIKIT-LEARN

SIMPLE REGRESSION ANALYSIS IN SCIKIT-LEARN

- scikit-learn defines models as **objects** (in the OOP sense)

- You can use the following principles
  - All scikit-learn modelling classes are based on the base estimator
    - This means all models take a similar form
  - All estimators take a Matrix **X**, either sparse or dense
  - Supervised estimators also take a Vector **y** (the response)
  - Estimators can be customised 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**, which are attributes or other information (Data that gives characteristics to the objects)
  - john_smith.age
  - john_smith.gender
- Object have **Methods**, which are procedures or functions or object (Code that gives behaviour to the objects)
  - john_smith.breathe()
  - john_smith.walk()

SIMPLE REGRESSION ANALYSIS IN SCIKIT-LEARN

- General format for scikit-learn 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() does not have a transform() function
- With this information, we can build a simple process for linear regression

SIGNIFICANCE IS KEY

DEMONSTRATION: 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

EXERCISE

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 below to complete #1

```
X =
y =
loop =
for boolean in loop:
  print "y-intercept:", boolean
  lm = linear_model.LinearRegression(fit_intercept = boolean)
  get_linear_model_metrics(X, y, lm)
  print
```

INDEPENDENT PRACTICE

BASE LINEAR REGRESSION CLASSES

ACTIVITY: BASE LINEAR REGRESSION CLASSES

EXERCISE

DIRECTIONS: (20 MINUTES)

1. Experiment with the model evaluation function we have (get_linear_model_metrics()) with the following scikit-learn estimator classes
   a. linear_model.Lasso()
   b. linear_model.Ridge()
   c. linear_model.ElasticNet()
2. **Note**: We will cover these new regression techniques in a later class

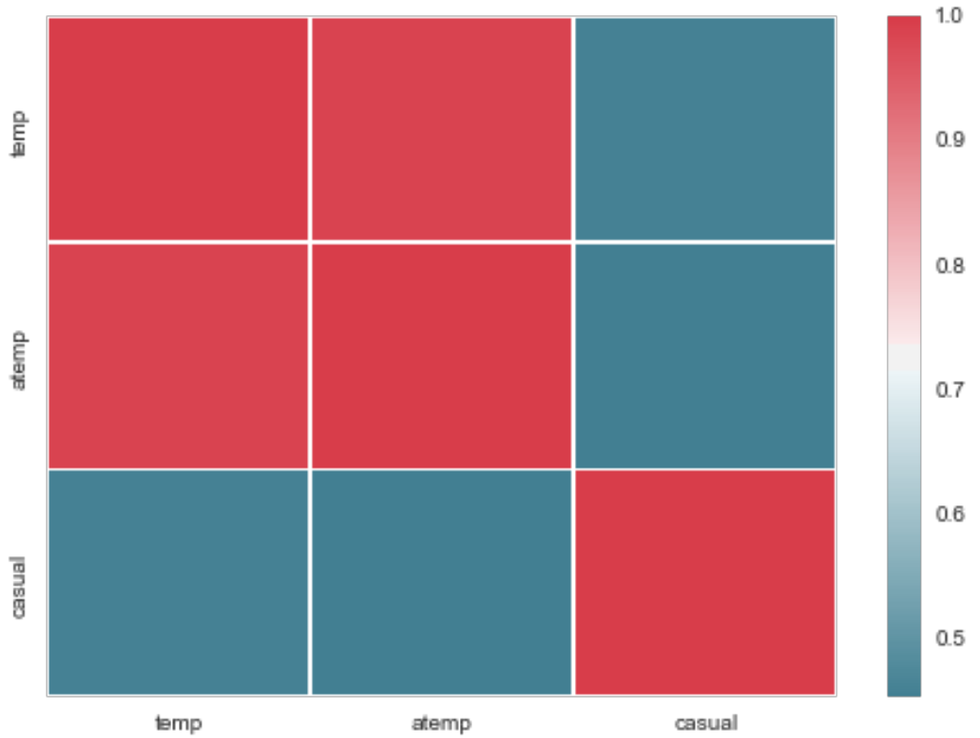
MULTIPLE REGRESSION ANALYSIS

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 is not highly correlated with temperature?



GUIDED PRACTICE

MULTICOLLINEARITY WITH DUMMY VARIABLES

ACTIVITY: MULTICOLLINEARITY WITH DUMMY VARIABLES

EXERCISE

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?

```
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)
```

GUIDED PRACTICE

COMBINING FEATURES INTO A BETTER MODEL

ACTIVITY: COMBINE FEATURES INTO A BETTER MODEL

EXERCISE

DIRECTIONS: (15 MINUTES)

1. With a partner, complete the code on the following slide
2. Visualise the correlations of all the numerical features built into the data set
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

```
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)
```

INDEPENDENT PRACTICE

BUILDING MODELS FOR OTHER Y VARIABLES

ACTIVITY: BUILDING MODELS FOR OTHER Y VARIABLES

DIRECTIONS (25 MINUTES)

1. 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?
   a. Can you build these features with the included data and pandas?

TOPIC REVIEW

TOPIC REVIEW

- 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?

BEFORE NEXT CLASS

BEFORE NEXT CLASS

DUE DATE

- Project
  - Final Project, part 1 is due on week 4, lesson 8

Q & A

EXIT TICKETS

DON'T FORGET TO FILL OUT YOUR EXIT TICKET

Exit Ticket Link

What's the lesson number?	06
What was the topic of the lesson?	Introduction to Regression Analysis