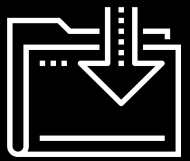
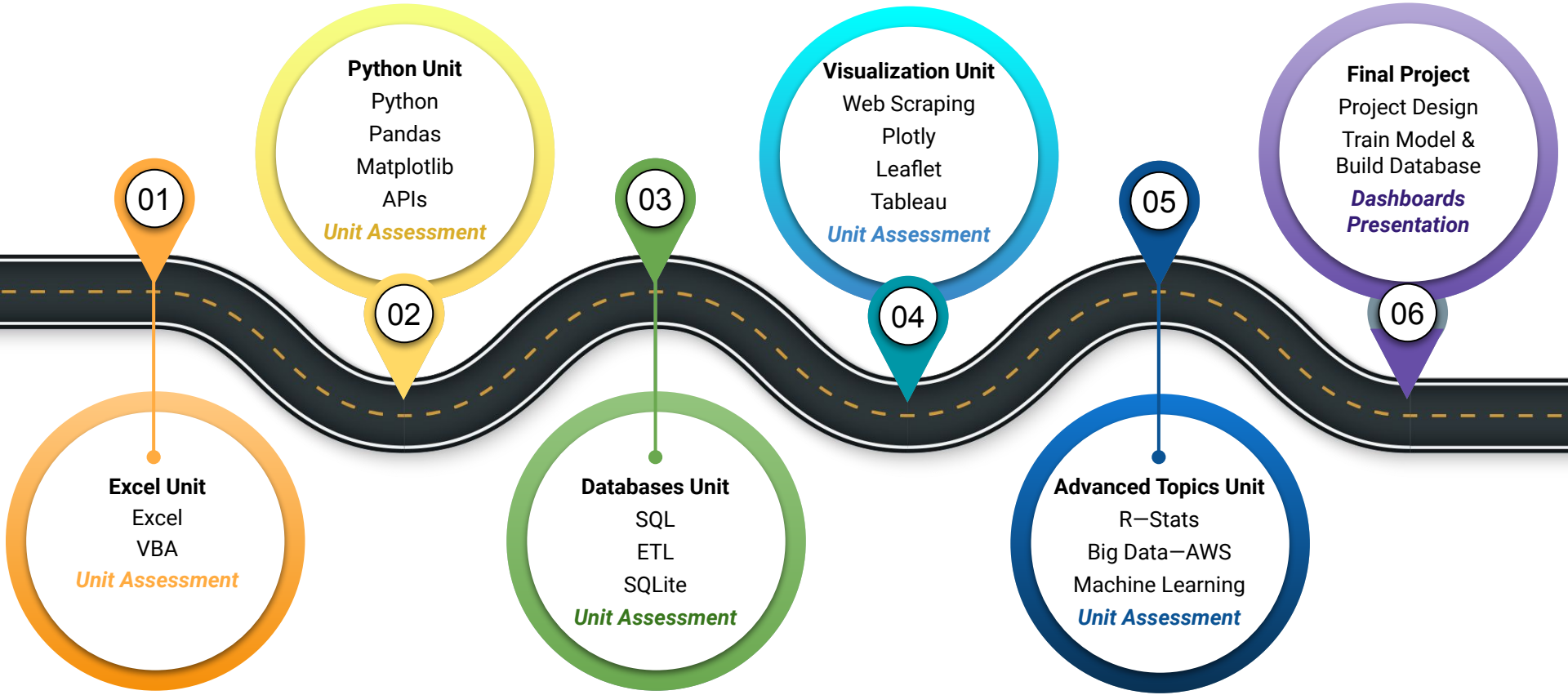
Capstone Project: Segment 2

Data Boot Camp

Lesson 20-2.2



The Big Picture

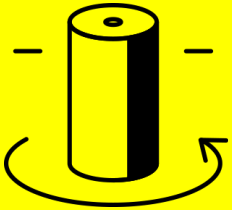


Project Segments

This Week: “Build The Pieces”

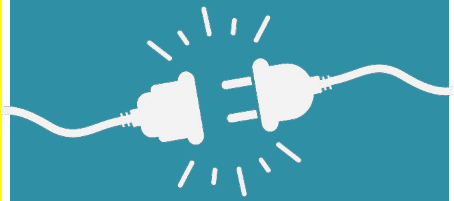
Sketch It Out

Decide on your overall project, select your question, and build a simple model. You'll connect the model to a fabricated database, using comma-separated values (CSV) or JavaScript Object Notation (JSON) files, to prototype your idea.



Build the Pieces

Train your model and build out the database you'll use for your final presentation.



Plug It In

Connect your final database to your model, continue to train your model, and create your dashboard and presentation.

Put It All Together

Put the final touches on your model, database, and dashboard. Lastly, create and deliver your final presentation to your class.

This Segment: Capstone Project

By the end of this segment, you'll will have:

Connected your machine learning model into the project

Optimizing the integration of the database into the project

Have all necessary GitHub branches merged

Finalized your dashboard

Module 20

Today’s Agenda

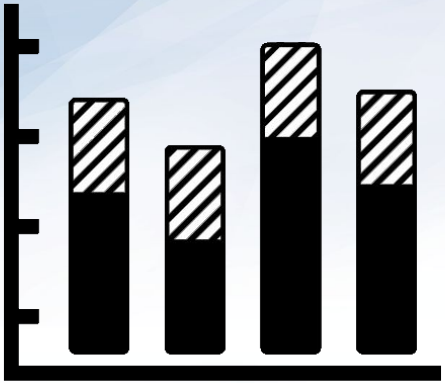
Today's Agenda

By completing today's activities, you'll...

01 Articulate the use cases for each machine learning model

02 Decide which machine learning model will fit their project best

Make sure you've downloaded any relevant class files!



Please respond to the following poll:

How are you feeling about the current state of your project?

1. It's ready for the next stage!
2. I'm a little stressed, but it will be ready for the next stage in time.
3. I'm really worried. Help!
4. I haven't started yet. Help!

Suggested Time:
5 minutes

Final Project: Session 2

Final Project: Session 2

At this stage in the project you should have the following completed:

01 Move from a preliminary model into your machine learning model.

02 The mockup database is integrated and refined.

03 The visuals that help tell the data story are created.

04 Merge in branches and create new ones for this segment's tasks.

05 Create an outline or storyboard for the final dashboard.

Group Project Check-In

Collaboration is key for team success.

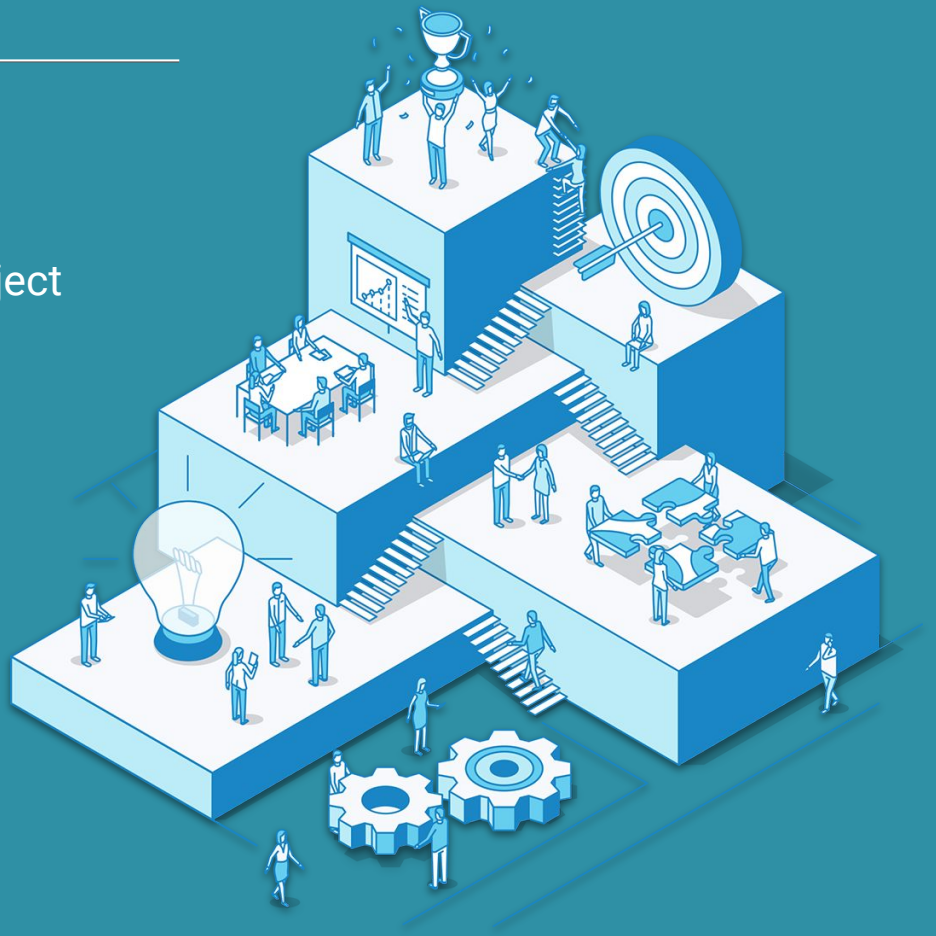
Assigning roles
Know who is working on what part of the project

Handling blockers
What work do you need others to finish before you can progress?

Are there database or ML issues?

Using Github
- Creating branches
- Reviewing pull requests

Plan what needs to be done by next week



Final Project: Session 2

Today's class will consist of the following:

Work on your project with your group.
TA's and myself will be circulating among the groups.

In the check-ins you will:

Show us your progress you have made as if we were shareholders.

In the check-ins we will:

- Answer questions.
- Help you work through sticking points.

Shareholder Update Pitch

We will spend some time with each group and review the progress you have made so far for your project as if we were shareholders for the business you work for.

This pitch /update should contain the following:

- A draft of the Google slides.
- GitHub repo and `README.md` file with description.
- A brief explanation of the description of ML model from preprocessing to training and testing.
- An overview of your database with tables and relationship, and a connection string (SQLAlchemy or PyMongo).
- A blueprint for the dashboard or storyboard (this can be part of the Google slides).
- Where you plan on being at by next week.

Choose Your Machine Learning Model

In this demonstration, we will go over steps to help you and your team decide if you need to pick a new model or optimize your current model.

Step 1: Analyze the input data.

If the input has data is labeled, choose a **supervised learning model**.

The Pima diabetes dataset had labeled data, so this was a good dataset for supervised learning.

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
0	6	148	72	35	0	33.6	0.627	50	1
1	1	85	66	29	0	26.6	0.351	31	0
2	8	183	64	0	0	23.3	0.672	32	1
3	1	89	66	23	94	28.1	0.167	21	0
4	0	137	40	35	168	43.1	2.288	33	1

Step 1: Analyze the input data.

If the input data is unlabeled, choose the **unsupervised learning model**.

The Boston marathon dataset was a good dataset to classify runners into age groups since there was no "age group" category.

```
Index(['Bib', 'Name', 'Age', 'M/F', 'City', 'State', 'Country', 'Citizen',
       'Unnamed: 8', '5K', '10K', '15K', '20K', 'Half', '25K', '30K', '35K',
       '40K', 'Pace', 'Proj Time', 'Official Time', 'Overall', 'Gender',
       'Division'],
      dtype='object')
```

Step 1: Analyze the input data.

If the input data contains a large number of variables or is in a non-tabular format, consider a **deep learning model**.

The IBM dataset we used to predict if an employee was at risk of attrition was a good example of a dataset with a large number of variables—with 35 columns.

```
Index(['Age', 'Attrition', 'BusinessTravel', 'DailyRate', 'Department',
       'DistanceFromHome', 'Education', 'EducationField', 'EmployeeCount',
       'EmployeeNumber', 'EnvironmentSatisfaction', 'Gender', 'HourlyRate',
       'JobInvolvement', 'JobLevel', 'JobRole', 'JobSatisfaction',
       'MaritalStatus', 'MonthlyIncome', 'MonthlyRate', 'NumCompaniesWorked',
       'Over18', 'OverTime', 'PercentSalaryHike', 'PerformanceRating',
       'RelationshipSatisfaction', 'StandardHours', 'StockOptionLevel',
       'TotalWorkingYears', 'TrainingTimesLastYear', 'WorkLifeBalance',
       'YearsAtCompany', 'YearsInCurrentRole', 'YearsSinceLastPromotion',
       'YearsWithCurrManager'],
      dtype='object')
```

Step 2: Analyze the output data.

If the output of the model should predict a value or outcome, **choose regression**.

For the Pima diabetes dataset, we used a regression model to predict outcome.

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
0	6	148	72	35	0	33.6	0.627	50	1
1	1	85	66	29	0	26.6	0.351	31	0
2	8	183	64	0	0	23.3	0.672	32	1
3	1	89	66	23	94	28.1	0.167	21	0
4	0	137	40	35	168	43.1	2.288	33	1

Step 2: Analyze the output data.

If the output of the model should be the name of a group or class, choose **classification**.

For the Boston marathon dataset, we used classification to group runners into age groups.

```
Index(['Bib', 'Name', 'Age', 'M/F', 'City', 'State', 'Country', 'Citizen',
       'Unnamed: 8', '5K', '10K', '15K', '20K', 'Half', '25K', '30K', '35K',
       '40K', 'Pace', 'Proj Time', 'Official Time', 'Overall', 'Gender',
       'Division'],
      dtype='object')
```

Step 2: Analyze the output data.

If the output of the model should be the name of a group or class, choose **classification**.

For the IBM dataset, we determined accuracy of the model so we can use it to classify whether an employee is at risk of attrition.

```
Index(['Age', 'Attrition', 'BusinessTravel', 'DailyRate', 'Department',
       'DistanceFromHome', 'Education', 'EducationField', 'EmployeeCount',
       'EmployeeNumber', 'EnvironmentSatisfaction', 'Gender', 'HourlyRate',
       'JobInvolvement', 'JobLevel', 'JobRole', 'JobSatisfaction',
       'MaritalStatus', 'MonthlyIncome', 'MonthlyRate', 'NumCompaniesWorked',
       'Over18', 'OverTime', 'PercentSalaryHike', 'PerformanceRating',
       'RelationshipSatisfaction', 'StandardHours', 'StockOptionLevel',
       'TotalWorkingYears', 'TrainingTimesLastYear', 'WorkLifeBalance',
       'YearsAtCompany', 'YearsInCurrentRole', 'YearsSinceLastPromotion',
       'YearsWithCurrManager'],
      dtype='object')
```

Step 2: Analyze the output data.

If the output of the model should identify closely related data points, choose **clustering**.

We could also use clustering to classify runners from the Boston marathon dataset into age groups because the data points are closely related.

```
Index(['Bib', 'Name', 'Age', 'M/F', 'City', 'State', 'Country', 'Citizen',
       'Unnamed: 8', '5K', '10K', '15K', '20K', 'Half', '25K', '30K', '35K',
       '40K', 'Pace', 'Proj Time', 'Official Time', 'Overall', 'Gender',
       'Division'],
      dtype='object')
```

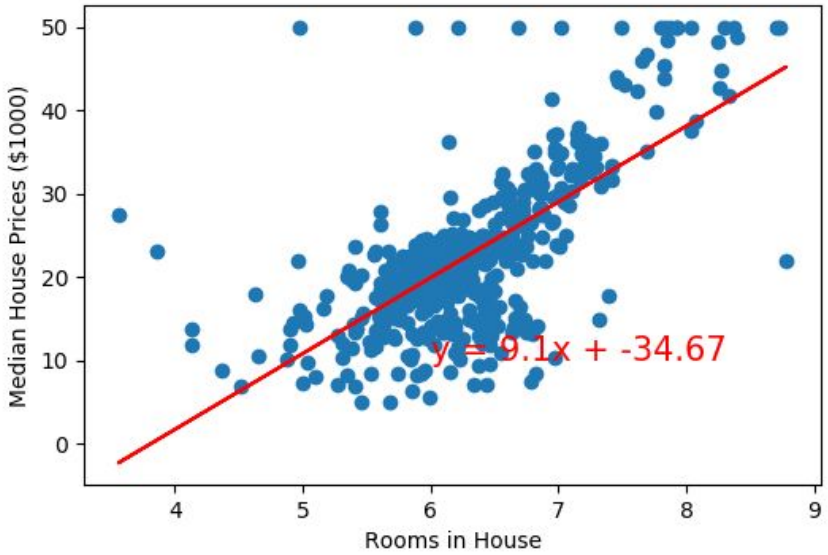
Once you have a firm idea of the data you have and what you hope to get out of it, it's time to look at what algorithm you will run.

Step 3: Choose your algorithm

Linear regression is used to predict continuous variables.

It will take in a set of factors and attempt to learn patterns from them to predict a numerical value.

If new data is added, the model will predict the result based on learned patterns—for example, determining the weight of a person based on caloric intake, height, and activity level.

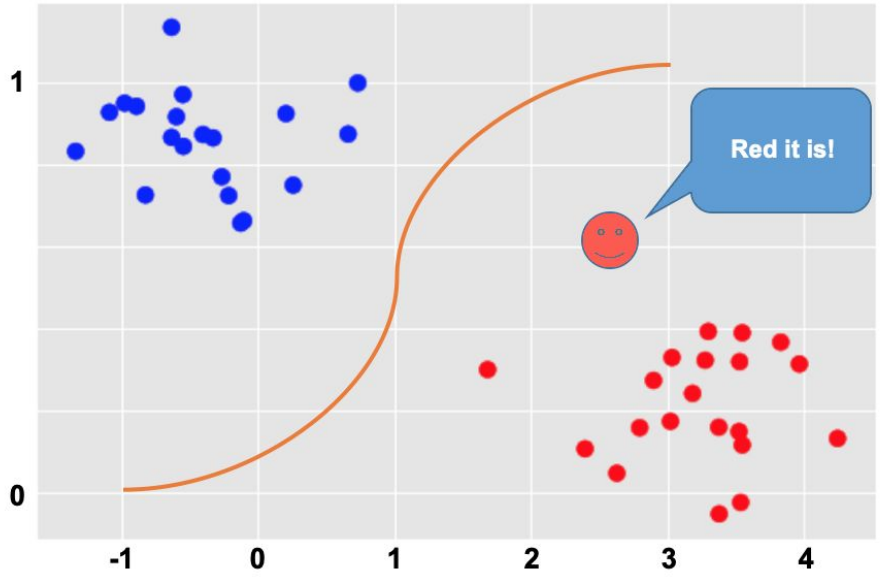


Step 3: Choose your algorithm

Logistic regression is typically used to predict binary outcomes, meaning that there are only two possible outcomes.

The model analyzes the available data and, when presented with a new sample, mathematically determines its probability of belonging to a class.

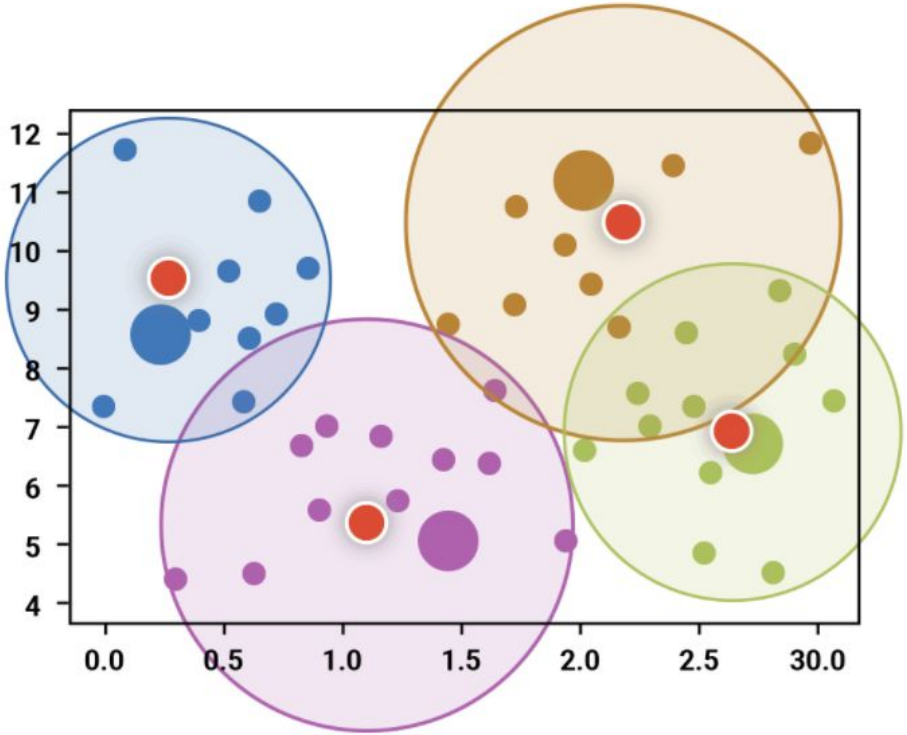
If the probability is above a certain cutoff point, the sample is assigned to that class. If the probability is below the cutoff point, the sample is assigned to the other class—for example, determining if a person will vote "Yes" or "No" on an issue based on things like income, location, and family size.



Step 3: Choose your algorithm

K-means is a clustering algorithm used to place data points into groups based on the distance between the points.

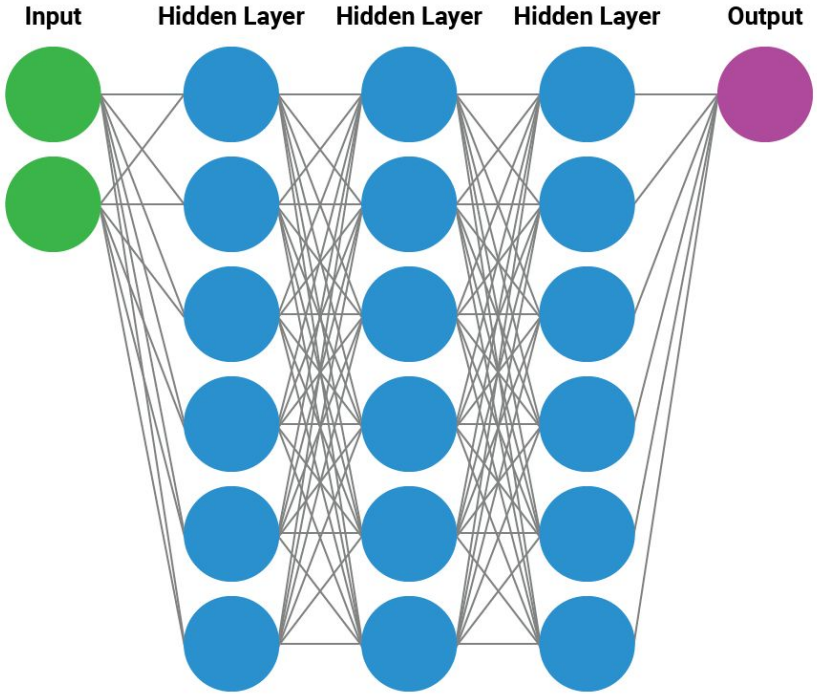
Points that belong to a cluster are more similar to each other than the points in another cluster—for example, grouping consumers based on income, time spent in store, amount spent, and age.



Step 3: Choose your algorithm

Neural networks are an advanced form of machine learning that can recognize patterns and features in input data and provide a clear quantitative output.

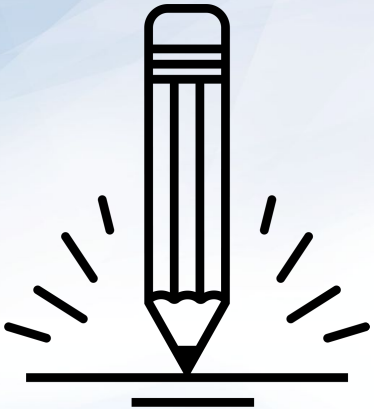
They can create a classification algorithm that determines if an input belongs in one category or another. Therefore, neural network models can be an alternative to many of the models we have learned throughout the course, including logistic regression and multiple linear regression.



Step 4: Analyze the results and review accuracy

Use the following questions to review the results of your output:

- How accurate was your model?
- Were the results easy to interpret, or are you still not sure what the result is?
- Did you choose the simplest model to achieve your goals, or did you overcomplicate things with a more advanced model?
- Will this model reproduce similar results each time it is run, or will the results differ drastically each time?



Activity: Choose Your Machine Learning Model

In this activity, you and your team can use supervised, unsupervised, or deep learning on your data to determine if you need to optimize or change your machine learning model for your final project.

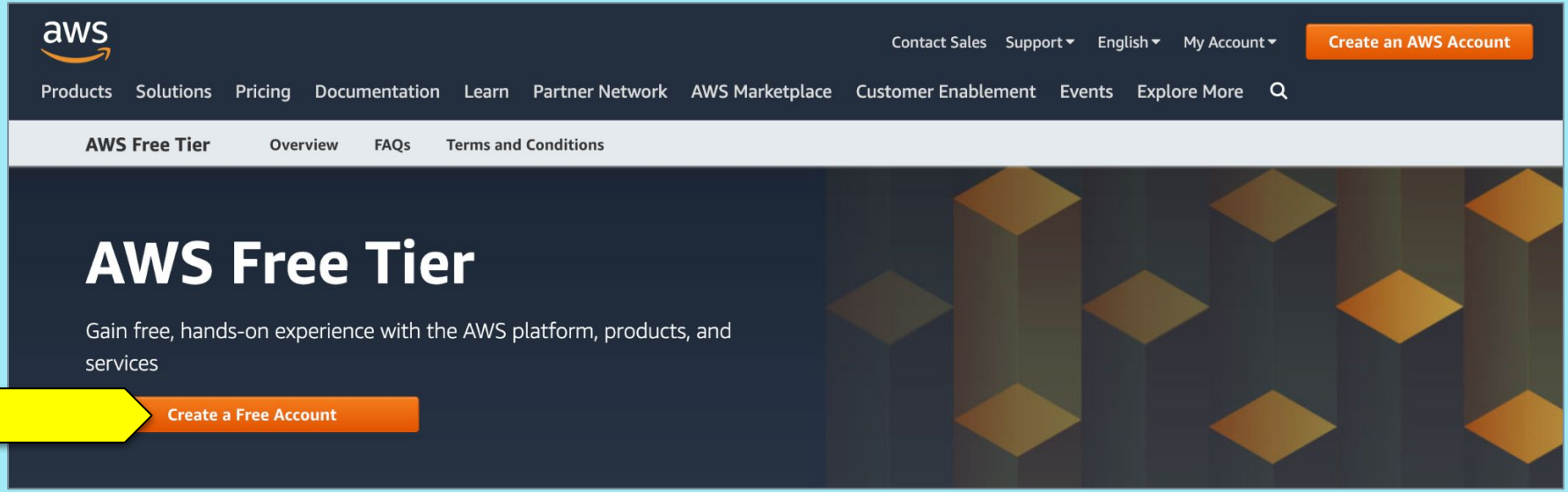
Suggested Time:
25 minutes

Project Progression—Next Week

Project Progression—Next Week

Next class, we will be creating and connecting to a PostgreSQL database in RDS on Amazon Web Services.

Please create a AWS Free Tier account before next class.

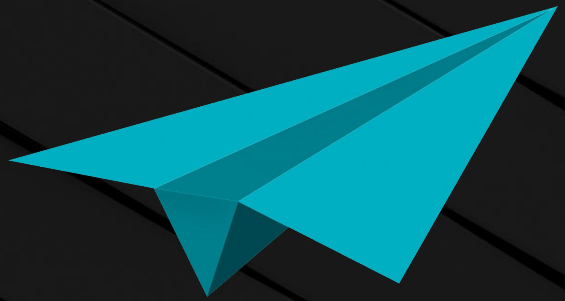


Project Progression—Next Steps

For the next class, you should be working on the following:

- Refining your machine learning model
- Integrating the database with the project
- Working on your dashboard
- Getting all the images you will need for your presentation
- Adding a project description in the GitHub repository `README.md`
- Cleaning up the GitHub repository
- Getting ready to add the repository to your portfolio

Questions?



Office Hours

30 minutes