Udacity Connect Intensive
Session 03 - Model Evaluation and Validation

October 29, 2016

Lutfur Khundkar

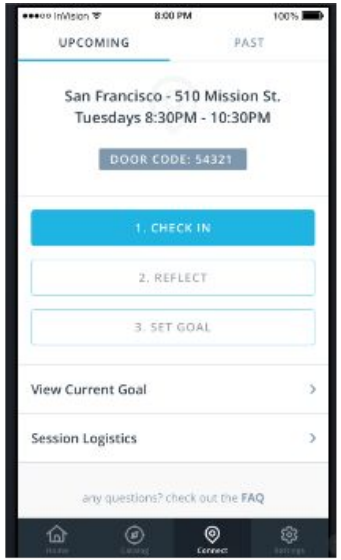
Outline - Morning

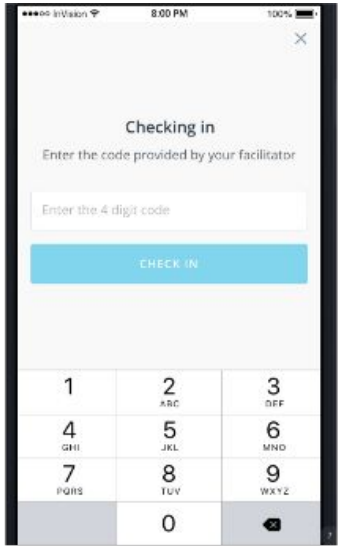
- Welcome back! (20 min)
  - Introduction
  - Check-in (via Udacity App)
  - Goal setting (via Udacity App)
  - Course schedule & this week's homework
- Jupyter Notebook Lesson (1 hr 20 min)
  - Practice using the sklearn library
- Review of Notebook Lesson (20 min)

Housekeeping Items

ATTENDANCE AND CHECKING IN

- Check in via the Udacity app
- This week's check-in code: ****
  - Cogswellguest/dr******16
- MUST let me know if you can't check in!

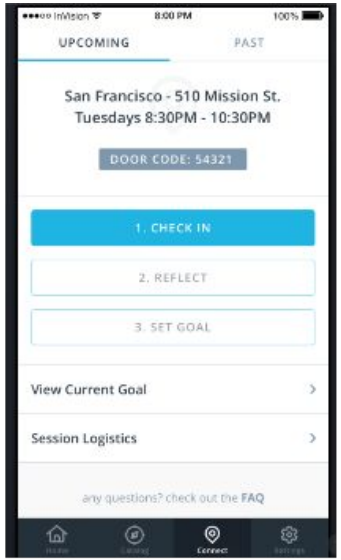


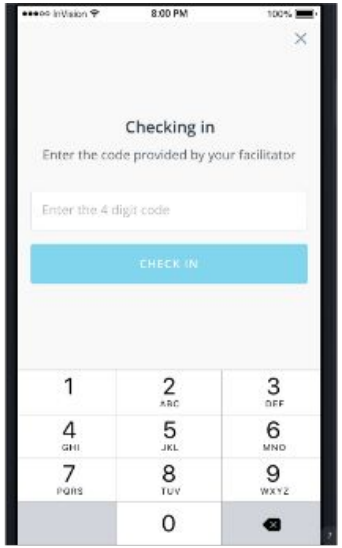


Housekeeping Items

GOAL SETTING - WEEK OF OCT 29

- "I will complete and submit the **Predicting Boston Housing Prices** project and report."
- "I will complete the lectures in the **Supervised Learning Tasks** section."
- "I will complete the lectures in the **Artificial Neural Networks** and **Bayesian Methods** sections, *saving the mini-projects for session next week*."





Connect Intensive

SCHEDULE: https://goo.gl/4D4EqL

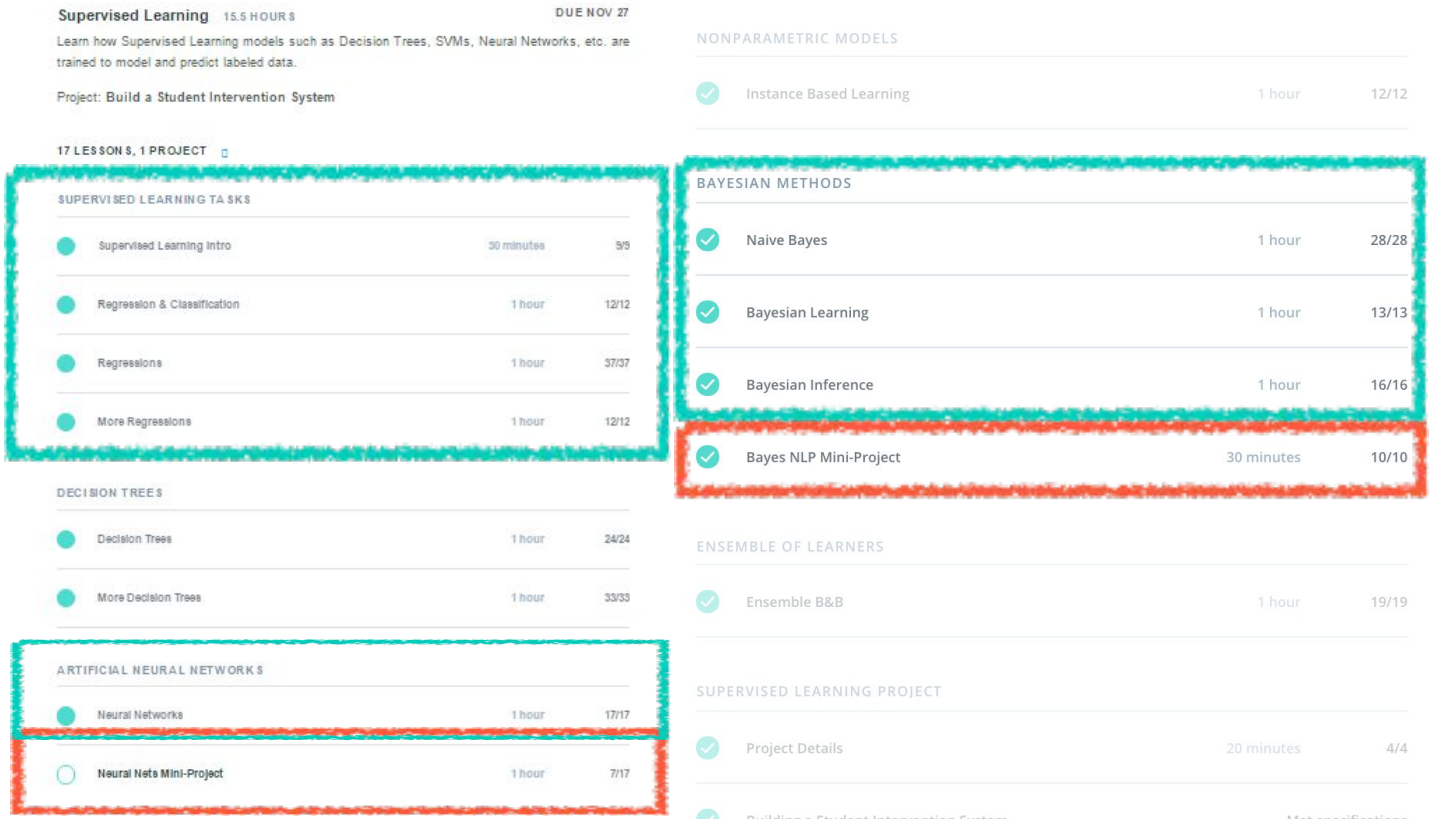
- Today's session:
  - Model evaluation and validation
- This week's homework lectures:
  - Supervised Learning Tasks
  - Artificial Neural Networks
  - Bayesian Methods
- Submit the Boston Housing Prices project and report this week!

Date	Session	Homework
OCT 15	Thinking Like a Machine Learnist	Begin the **Model Evaluation and Validation** course. Complete sections: • Statistical Analysis • Data Modeling: **do not** do the section in Data Modeling titled "Datasets and Questions"; this will be covered in session next week.
OCT 22	Data Modeling with the Enron scandal dataset	Finish the **Model Evaluation and Validation** course. And review the **Predicting Boston Housing Prices** project. The following sections from the course should be completed: • Evaluation and Validation • Managing Error and Complexity
OCT 29	Model evaluation and validation	Complete and submit the **Predicting Boston Housing Prices** project (report). Begin the **Supervised Learning** course. The following sections from the course should be completed: • Supervised Learning Tasks • Artificial Neural Networks: **do not do Neural Nets Mini-Project**; this will be covered in session next week • Bayesian Methods: **do not do Bayes NLP Mini-Project**; this will be covered in session next week Note that the itinerary for this week is not in sync with that of the current Nanodegree Syllabus.
NOV 05	Natural Language and Processing and Neural Networks	Finish the **Supervised Learning** course and review the **Building a Student Intervention System** project. The following sections from the course should be completed: • Decision Trees • Support Vector Machines • Nonparametric Models • Ensemble of Learners Note that the itinerary for this week is not in sync with that of the current Nanodegree Syllabus.

This week's homework lectures

lectures to complete before session next week

mini projects to do in session next week



This Weeks Homework (readable)

Supervised Learning

Supervised Learning Tasks

- Supervised Learning Intro
- Regression and Classification
- Regression
- More Regression

Decision Trees -- defer

Artificial Neural Networks

- Neural Networks
- NN Mini project -- defer

Other sections - defer

Bayesian Networks

- Naive Bayes
- Bayesian Learning
- Bayesian Inference
- Bayes NPL mini project - defer

Jupyter Notebook Lesson

PRACTICE WITH sklearn LIBRARY

- Exercises:
  https://github.com/nickypie/ConnectIntensive
  lesson-03-part02.ipynb and lesson-03-part02.ipynb
- Solutions in solutions-03.ipynb (for part 01)

The scikit-learn (sklearn) library:

- Installation: http://scikit-learn.org/stable/install.html
- **NOTE:** The latest stable release of sklearn is 0.18 (September 28, 2016)
- The module model_selection (0.18) groups together the functionalities of:
  - cross_validation, grid_search, learning_curve (0.17)
- For more info: http://scikit-learn.org/stable/whats_new.html#version-0-18

0.17	0.18
sklearn.cross_validation.train_test_split	sklearn.model_selection.train_test_split
sklearn.cross_validation.ShuffleSplit	sklearn.model_selection.ShuffleSplit
sklearn.grid_search.GridSearchCV	sklearn.model_selection.GridSearchCV
sklearn.learning_curve.learning_curve	sklearn.model_selection.learning_curve

Data

Exploration, pre-processing

Data Types

- Three types of data - numerical, categorical, time series

Variability of Data

- DIfferent ways of quantifying the variability (often good to visualize)
- Underlying causes
- Random or measurement error

The goal of fitting a model to any data is to capture and “explain away” the variance in the data due to causes

Should not try to account for random error -- leads to overfitting

Model

Model Building -- key points to keep in mind as you follow supervised learning videos

Bias vs Variance

Underfitting vs Overfitting

Curse of Dimensionality

Validation

Model Evaluation and Validation

Performance Metric

Test *vs*. Training Error

Cross Validation

Model Fitting with sklearn

Generic plan

Import data – features and labels

Split data into test and train sets

Import library function, e.g. `from sklearn.tree import *Classifier`

Create classifier, set parameters for model

Fit(train_features, train_labels)

Predict(test_features)

Score

Evaluation Metrics

Classification

Two common metrics:

- accuracy (may not be appropriate for skewed classes)
- F1 score = 2 * precision * recall /(precision + recall)

A more general way to look at results is the confusion matrix

- Precision and recall are two among many different terms used in practice
- The term being used often depends on the problem domain

Precision and Recall

		Predicted condition			
	Total population	Predicted Condition positive	Predicted Condition negative	Prevalence $= \frac{\Sigma \text{ Condition positive}}{\Sigma \text{ Total population}}$	
True condition	condition positive	**True positive**	**False Negative** (Type II error)	True positive rate (TPR), Sensitivity, Recall $= \frac{\Sigma \text{ True positive}}{\Sigma \text{ Condition positive}}$	False negative rate (FNR) Miss rate $= \frac{\Sigma \text{ False negative}}{\Sigma \text{ Condition positive}}$
	condition negative	**False Positive** (Type I error)	**True negative**	False positive rate (FPR), Fall-out $= \frac{\Sigma \text{ False positive}}{\Sigma \text{ Condition negative}}$	True negative rate (TNR), Specificity (SPC) $= \frac{\Sigma \text{ True negative}}{\Sigma \text{ Condition negative}}$
	Accuracy (ACC) $= \frac{\Sigma \text{ True positive} + \Sigma \text{ True negative}}{\Sigma \text{ Total population}}$	Positive predictive value (PPV), Precision $= \frac{\Sigma \text{ True positive}}{\Sigma \text{ Test outcome positive}}$	False omission rate (FOR) $= \frac{\Sigma \text{ False negative}}{\Sigma \text{ Test outcome negative}}$	Positive likelihood ratio (LR+) $= \frac{\text{TPR}}{\text{FPR}}$	Diagnostic odds ratio (DOR) $\frac{\text{LR+}}{\text{LR−}}$
		False discovery rate (FDR) $= \frac{\Sigma \text{ False positive}}{\Sigma \text{ Test outcome positive}}$	Negative predictive value (NPV) $= \frac{\Sigma \text{ True negative}}{\Sigma \text{ Test outcome negative}}$	Negative likelihood ratio (LR−) $= \frac{\text{FNR}}{\text{TNR}}$	

https://en.wikipedia.org/wiki/Precision_and_recall

Evaluation Metrics

Regression

Two common measures of error - smaller is better

- Mean absolute error
- Mean squared error

Two common scoring methods - higher is better

- R2-score
- Explained variance

Sklearn classifiers and regressors general have a default scoring metric to optimize the fit that can be overriden.