Project Overview

In this project, you will play detective, and put your machine learning skills to use by building an algorithm to identify Enron Employees who may have committed fraud based on the public Enron financial and email dataset.

Why this Project?

This project will teach you the end-to-end process of investigating data through a machine learning lens.

It will teach you how to extract/identify useful features that best represents your data, a few of the most commonly used machine learning algorithms today, and how to evaluate the performance of your machine learning algorithms.

What will I learn?

By the end of the project, you will be able to:

- Deal with an imperfect, real-world dataset
- Validate a machine learning result using test data
- Evaluate a machine learning result using quantitative metrics
- Create, select and transform features compare the performance of machine learning algorithms
- Tune machine learning algorithms for maximum performance
- Communicate your machine learning algorithm results clearly

Why is this Important to my Career?

Machine Learning is a first-class ticket to the most exciting careers in data analysis today.

As data sources proliferate along with the computing power to process them, going straight to the data is one of the most straightforward ways to quickly gain insights and make predictions.

Machine learning brings together computer science and statistics to harness that predictive power.

NEXT

Project Introduction

In 2000, Enron was one of the largest companies in the United States. By 2002, it had collapsed into bankruptcy due to widespread corporate fraud. In the resulting Federal investigation, there was a significant amount of typically confidential information entered into public record, including tens of thousands of emails and detailed financial data for top executives. In this project, you will play detective, and put your new skills to use by building a person of interest identifier based on financial and email data made public as a result of the Enron scandal. To assist you in your detective work, we've combined this data with a hand-generated list of persons of interest in the fraud case, which means individuals who were indicted, reached a settlement, or plea deal with the government, or testified in exchange for prosecution immunity.

Pep Talk

A note before you begin: the projects in the Intro to Machine Learning class were mostly designed to have lots of data points, give intuitive results, and otherwise behave nicely. This project is significantly tougher in that we're now using the real data, which can be messy and doesn't have as many data points as we usually hope for when doing machine learning. Don't get discouraged--imperfect data is something you need to be used to as a data analyst! If you encounter something you haven't seen before, take a step back and think about a smart way around. You can do it!

Resources Needed

You should have python and sklearn running on your computer, as well as the starter code (both python scripts and the Enron dataset) that you downloaded as part of the first mini-project in the Intro to Machine Learning course. The starter code can be found in the final_project directory of the codebase that you downloaded for use with the mini-projects. Some relevant files:

poi_id.py : starter code for the POI identifier, you will write your analysis here

final_project_dataset.pkl : the dataset for the project, more details below

tester.py : when you turn in your analysis for evaluation by a Udacity evaluator, you will submit the algorithm, dataset and list of features that you use (these are created automatically in poi_id.py). The evaluator will then use this code to test your result, to make sure we see performance that's similar to what you report. You don't need to do anything with this code, but we provide it for transparency and for your reference.

emails_by_address : this directory contains many text files, each of which contains all the messages to or from a particular email address. It is for your reference, if you want to create more advanced features based on the details of the emails dataset.

Steps to Success

We will provide you with starter code, that reads in the data, takes your features of choice, then puts them into a numpy array, which is the input form that most sklearn functions assume. Your job is to engineer the features, pick and tune an algorithm, test, and evaluate your identifier. Several of the mini-projects were designed with this final project in mind, so be on the lookout for ways to use the work you've already done.

The features in the data fall into three major types, namely financial features, email features and POI labels.

- financial features: ['salary', 'deferral_payments', 'total_payments', 'loan_advances', 'bonus', 'restricted_stock_deferred', 'deferred_income', 'total_stock_value', 'expenses', 'exercised_stock_options', 'other', 'long_term_incentive', 'restricted_stock', 'director_fees'] (all units are in US dollars)
- email features: ['to_messages', 'email_address', 'from_poi_to_this_person', 'from_messages', 'from_this_person_to_poi', 'shared_receipt_with_poi'] (units are generally number of emails messages; notable exception is 'email_address', which is a text string)
- POI label: ['poi'] (boolean, represented as integer)

You are encouraged to make, transform or rescale new features from the starter features. If you do this, you should store the new feature to my_dataset, and if you use the new feature in the final algorithm, you should also add the feature name to my_feature_list, so your coach can access it during testing. For a concrete example of a new feature that you could add to the dataset, refer to the lesson on Feature Selection.

NEXT

Enron Submission Free-Response Questions

A critical part of machine learning is making sense of your analysis process and communicating it to others. The questions below will help us understand your decision-making process and allow us to give feedback on your project. Please answer each question; your answers should be about 1-2 paragraphs per question. If you find yourself writing much more than that, take a step back and see if you can simplify your response!

When your evaluator looks at your responses, he or she will use a specific list of rubric items to assess your answers. Here is the link to that rubric: [**Link**] Each question has one or more specific rubric items associated with it, so before you submit an answer, take a look at that part of the rubric. If your response does not meet expectations for all rubric points, you will be asked to revise and resubmit your project. Make sure that your responses are detailed enough that the evaluator will be able to understand the steps you took and your thought processes as you went through the data analysis.

Once you've submitted your responses, your coach will take a look and may ask a few more focused follow-up questions on one or more of your answers.

We can't wait to see what you've put together for this project!

1. Summarize for us the goal of this project and how machine learning is useful in trying to accomplish it. As part of your answer, give some background on the dataset and how it can be used to answer the project question. Were there any outliers in the data when you got it, and how did you handle those? [relevant rubric items: "data exploration", "outlier investigation"]

2. What features did you end up using in your POI identifier, and what selection process did you use to pick them? Did you have to do any scaling? Why or why not? As part of the assignment, you should attempt to engineer your own feature that does not come ready-made in the dataset -- explain what feature you tried to make, and the rationale behind it. (You do not necessarily have to use it in the final analysis, only engineer and test it.) In your feature selection step, if you used an algorithm like a decision tree, please also give the feature importances of the features that you use, and if you used an automated feature selection function like SelectKBest, please report the feature scores and reasons for your choice of parameter values. [relevant rubric items: "create new features", "intelligently select features", "properly scale features"]

3. What algorithm did you end up using? What other one(s) did you try? How did model performance differ between algorithms? [relevant rubric item: "pick an algorithm"]

4. What does it mean to tune the parameters of an algorithm, and what can happen if you don’t do this well? How did you tune the parameters of your particular algorithm? What parameters did you tune? (Some algorithms do not have parameters that you need to tune -- if this is the case for the one you picked, identify and briefly explain how you would have done it for the model that was not your final choice or a different model that does utilize parameter tuning, e.g. a decision tree classifier). [relevant rubric items: “discuss parameter tuning”, “tune the algorithm”]

5. What is validation, and what’s a classic mistake you can make if you do it wrong? How did you validate your analysis? [relevant rubric items: “discuss validation”, “validation strategy”]

6. Give at least 2 evaluation metrics and your average performance for each of them. Explain an interpretation of your metrics that says something human-understandable about your algorithm’s performance. [relevant rubric item: “usage of evaluation metrics”]

PROJECT SPECIFICATION

Identify Fraud from Enron Email

Quality of Code

CRITERIA	MEETS SPECIFICATIONS
Functionality	Code reflects the description in the answers to questions in the writeup. i.e. code performs the functions documented in the writeup and the writeup clearly specifies the final analysis strategy.
Usability	poi_id.py can be run to export the dataset, list of features and algorithm, so that the final algorithm can be checked easily using tester.py.

Understanding the Dataset and Question

CRITERIA	MEETS SPECIFICATIONS
Data Exploration (related lesson: "Datasets and Questions")	Student response addresses the most important characteristics of the dataset and uses these characteristics to inform their analysis. Important characteristics include: • total number of data points • allocation across classes (POI/non-POI) • number of features used • are there features with many missing values? etc.
Outlier Investigation (related lesson: "Outliers")	Student response identifies outlier(s) in the financial data, and explains how they are removed or otherwise handled.

Optimize Feature Selection/Engineering

CRITERIA	MEETS SPECIFICATIONS
Create new features (related lesson: "Feature Selection")	At least one new feature is implemented. Justification for that feature is provided in the written response. The effect of that feature on final algorithm performance is tested or its strength is compared to other features in feature selection. The student is not required to include their new feature in their final feature set.
Intelligently select features (related lesson: "Feature Selection")	Univariate or recursive feature selection is deployed, or features are selected by hand (different combinations of features are attempted, and the performance is documented for each one). Features that are selected are reported and the number of features selected is justified. For an algorithm that supports getting the feature importances (e.g. decision tree) or feature scores (e.g. SelectKBest), those are documented as well.
Properly scale features (related lesson: "Feature Scaling")	If algorithm calls for scaled features, feature scaling is deployed.

Pick and Tune an Algorithm

CRITERIA	MEETS SPECIFICATIONS
Pick an algorithm (related lessons: "Naive Bayes" through "Choose Your Own Algorithm")	At least two different algorithms are attempted and their performance is compared, with the best performing one used in the final analysis.
Discuss parameter tuning and its importance.	Response addresses what it means to perform parameter tuning and why it is important.
Tune the algorithm (related lesson: "Validation")	At least one important parameter tuned with at least 3 settings investigated systematically, or any of the following are true: • GridSearchCV used for parameter tuning • Several parameters tuned • Parameter tuning incorporated into algorithm selection (i.e. parameters tuned for more than one algorithm, and best algorithm-tune combination selected for final analysis).

Validate and Evaluate

CRITERIA	MEETS SPECIFICATIONS
Usage of Evaluation Metrics (related lesson: "Evaluation Metrics")	At least two appropriate metrics are used to evaluate algorithm performance (e.g. precision and recall), and the student articulates what those metrics measure in context of the project task.
Discuss validation and its importance.	Response addresses what validation is and why it is important.
Validation Strategy (related lesson "Validation")	Performance of the final algorithm selected is assessed by splitting the data into training and testing sets or through the use of cross validation, noting the specific type of validation performed.
Algorithm Performance	When tester.py is used to evaluate performance, precision and recall are both at least 0.3.