

DS 5220

Supervised Machine Learning

Steve Morin

Project Discussion

Project Introduction

Adapted from Geron Chapter 2 - End-to-End Machine Learning Project

Groups are allowed with 3 people maximum per group.

Working with Real Data

Popular data repositories:

- OpenML.org (<https://openml.org>)
- Kaggle.com (<https://www.kaggle.com/datasets>)
- PaperWithCode (<https://paperswithcode.com/datasets>)
- UC Irvine Machine Learning Repository (<https://archive.ics.uci.edu/datasets>)
- Amazon's AWS datasets (<https://registry.opendata.aws>)
- TensorFlow datasets (<https://www.tensorflow.org/datasets>)

Project Phases

				% of Final Grade
Project Phase	Project Phase Description	Assign Date	Due Date	
1	Project Proposal	9/12/23	9/24/23	5
2	Project Progress Report	9/12/23	10/29/23	10
3	Project Final Report	9/12/23	12/10/23	15

Main Steps

- I. Frame the problem and look at the big picture.
- II. Get the data.
- III. Explore the data to get insights (EDA).
- IV. Prepare the data to better expose the underlying data patterns to machine learning algorithms.
- V. Explore many different models and short list the best ones.
- VI. Fine-tune your model.
- VII. Present your solution.



Minimum requirements for Project Phase 1
(Steps I and II)

Minimum requirements for Project Phase 2
(Steps III and IV)

Minimum requirements for Project Phase 3
(Steps V, VI and VII)

Project Deliverables

Project Phase	Steps	Deliverables
1	I and II	A PowerPoint document submitted as a .pdf file that addresses all the points in these steps as described in this document.
2	III and IV	<p><u>Step III:</u></p> <ul style="list-style-type: none">• One Jupyter notebook (eda.ipynb and eda.html) dedicated to data exploration as described in this document.• All .py modules that are imported by the notebooks.• The .yaml for the environment in which the data exploration was completed.• A PowerPoint document submitted as a .pdf file that describes the findings. <p><u>Step IV:</u></p> <ul style="list-style-type: none">• One Jupyter notebook (prep.ipynb and prep.html) dedicated to data preparation as described in this document.• All .py modules that are imported by the notebooks.• The .yaml for the environment in which the data preparation was completed.• A PowerPoint document submitted as a .pdf file that describes the data preparation.

Project Deliverables (continued)

Project Phase	Steps	Deliverables
3	V, VI and VII	<p><u>Step V:</u></p> <ul style="list-style-type: none">• Jupyter notebook(s) (model_exp_x.ipynb and model_exp_x.html) dedicated to “quick and dirty” model exploration as described in this document.• All .py modules that are imported by the notebooks.• The .yml for the environment in which the model exploration was completed.• A PowerPoint document submitted as a .pdf file that describes the model exploration and provides a short list of candidate models. <p><u>Step VI:</u></p> <ul style="list-style-type: none">• Jupyter notebook(s) (model_ft_x.ipynb and model_ft_x.html) dedicated to fine tuning a short list of candidate models as described in this document.• All .py modules that are imported by the notebooks.• The .yml for the environment in which the model fine tuning was completed.• A PowerPoint document submitted as a .pdf file that describes the model fine tuning and indicates the best model. <p><u>Step VII:</u></p> <ul style="list-style-type: none">• A PowerPoint document submitted as a .pdf file that addresses all the points in this step as described in this document.

Notes:

1. A file name that ends in a _x is meant to enable multiple notebooks in a submission. For example, for Step VI, one might submit the following: model_ft_1.ipynb and model_ft_2.ipynb.

I. Frame the Problem and Look at the Big Picture

1. Define the objective in business terms.
2. How will your solution be used?
3. How should performance be measured?
4. Is the performance measure aligned with the business objective?
5. What would be the minimum performance needed to reach the business objective?
6. List the assumptions you have made so far.
7. Verify assumptions if possible.

II. Get the Data

1. Find and document where you got the data.
2. Get the data.
3. Convert the data to a format you can easily manipulate (without changing the data itself).
4. Sample a test set, put it aside, and never look at it to avoid data leakage through the data scientist.

III. Explore the Data

1. Create a copy of the data for exploration.
2. Create a dedicated Jupyter notebook to keep a record of your data exploration.
3. Study each attribute and its characteristics:
 - a. Attribute name
 - b. Attribute type (categorical or numerical)
 - c. % of missing values
 - d. Boxplot showing outliers
 - e. Usefulness for the task
 - f. If numerical histogram and basic stats (mean, median, variance, max, min). If categorical, cardinality and value counts.
4. Identify the target attribute.
5. Additional visualization of the data as needed.


III. Explore the Data (continued)

6. Study the correlations and associations between attributes.
7. Study how you would solve the problem manually.
8. Identify the promising transformations you may want to apply.
9. Identify additional data that would be useful (go back to “Get the Data”).
10. Document what you have learned.

IV. Prepare the Data

Notes:

- Work on copies of the data. Keep the original data intact.
- Properly implement all data transformations you apply for reusability, for five reasons:
 - So you can easily prepare the data the next time you get a fresh data set.
 - So you can apply these transformations in future projects.
 - To clean and prepare the test set.
 - To clean and prepare new data instances once your solution is being used.
 - To make it easy to treat your preparation choices as hyperparameters.



sklearn pipelines will enable you to do this very easily!

IV. Prepare the Data (continued)

Notes (continued):

- Create a dedicated Jupyter notebook to keep a record of your data preparation.
-
1. Clean the data:
 - a. Fix or remove outliers (optional).
 - b. Fill in missing values or drop their rows (or columns).
 2. Perform feature selection (optional).
 - a. Drop the attributes that provide no useful information for the task.

IV. Prepare the Data (continued)

1. Perform feature engineering where appropriate.
 - a. Discretize continuous features.
 - b. Decompose features (for example, categorical, date/time, etc.).
 - c. Add promising transformations of features (for example, $\log(x)$, \sqrt{x} , x^2 , etc.).
 - d. Aggregate features into promising new features.
 - e. Perform feature scaling:
 - Standardize or normalize features.

V. Shortlist Promising Models

1. Train many quick and dirty models from different categories (linear, ensemble, neural net, etc.) using standard parameters.
2. Measure and compare their performance:
 - a. For each model, use N-fold cross-validation and compute the mean and standard deviation of the performance measure on the N folds.

Notes:

- If the data is huge sample smaller training sets so you can train many models. This will penalize complex models such as large neural nets or random forests.
- Try to automate these steps as much as possible.

V. Shortlist Promising Models (continued)

3. Analyze the most significant attributes for each algorithm.
4. Analyze the types of errors the models make:
 - a. What data would a human have used to avoid these errors?
5. Perform a quick round of feature selection and engineering.
6. Perform one or two more quick iterations of the five previous steps.
7. Shortlist the top three to five most promising models, preferring models that make different types of errors.

VI. Fine-tune the System

1. Fine-tune the hyperparameters using cross validation.
 - a. Treat your data transformation choices as hyperparameters.
 - b. Use random search over grid search. For long training runs you may want to use a Bayesian optimization approach.
2. Try model ensemble methods. Combining your best models will often produce better results than running them individually.
3. Once you are confident about your final model, measure its performance on the test set to estimate the generalization error.

Notes:

- Use as much data as possible for these steps.
- Try to automate these steps as much as possible.
- Don't tweak your model after measuring the generalization error: you would just start overfitting the test set.

VII. Present Your Solution

1. Document what you have done.
2. Create a presentation:
 - a. Highlight the big picture first.
3. Explain why your solution achieves the business objective.
4. Present interesting points you noticed along the way:
 - a. Describe what worked and what did not work.
 - b. List the assumptions and your systems limitations.

VII. Present Your Solution (continued)

5. Ensure your key findings are communicated through visualizations and easy to remember statements.