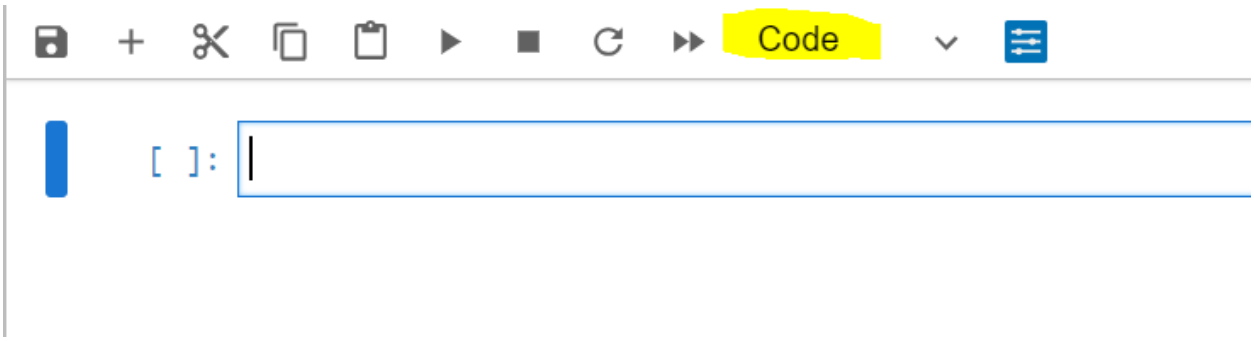
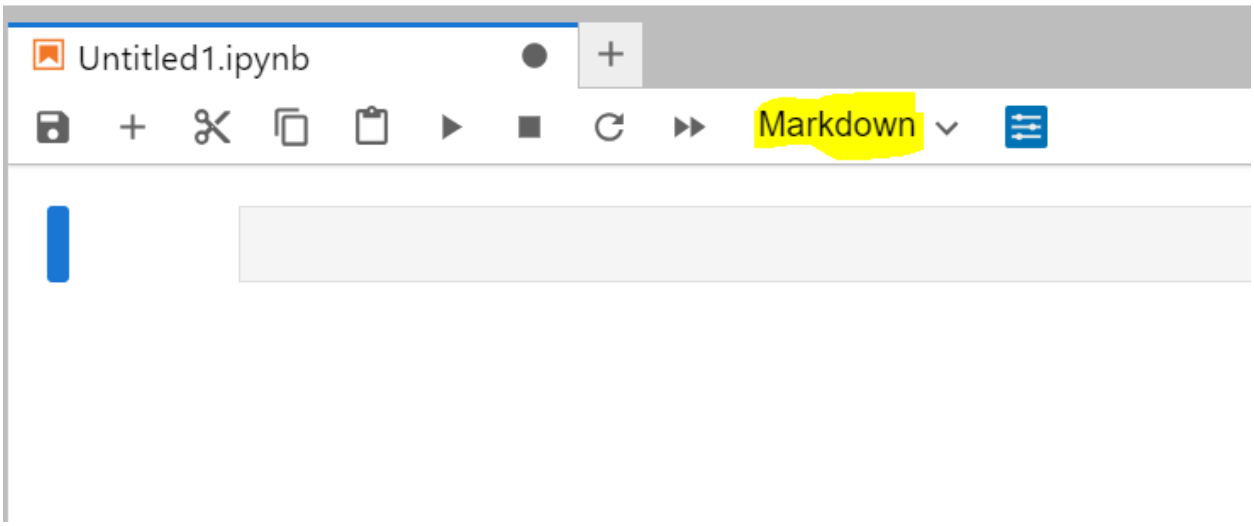
Final Project 15% of the course

Students will use Python programming language and it's libraries such as ***pandas, altair, sklearn and numpy*** to wrangle and transform the data provided and answer a predictive question about the given dataset.

Students should be creating a full analysis from the beginning to end (communicating their methods and conclusions).

In the Jupyter notebook, the **code cells** will read in the dataset, transform the data, create an estimator, and visualize the data. **Markdown cells** will be used throughout the document to narrate the analysis to communicate the question asked, methods used, and the conclusion reached.





Data description and important things to know.

For this project, students will be using a dataset regarding the different types of Canadian cheeses. The original data was found on the Government of Canada's Open Government Portal but has unfortunately been taken down. Given is the

modified version of the dataset. The data is stored in cheese_data.csv obtained from Kaggle.

The following columns have been included but you are **NOT required to use all of them!** I am including some instructions and suggestions on how to approach certain columns.

Column Name	Instructions
CheeseId	Drop this column however, you may also set it as your index if you so desire. This does not contribute to the prediction.
ManufacturerProvCode	
ManufacturingTypeEn	
MoisturePercent	
FlavourEn	This column could be transformed with CountVectorizer, however, this may be above the level of this course. You can drop this column or if you want to challenge yourself, you can leave it in and transform it accordingly.
CharacteristicsEn	This column could be transformed with CountVectorizer, however, this may be above the level of this course. You can drop this column or if you want to challenge yourself, you can leave it in and transform it accordingly.
Organic	
CategoryTypeEn	
MilkTypeEn	
MilkTreatmentTypeEn	
RindTypeEn	Do not use this column in analysis for convenience.
CheeseName	This column could be transformed with CountVectorizer, however, this may be above the level of this course. You can drop this column or if you want to challenge yourself, you can leave it in and transform it accordingly.
FatLevel	A suitable column to use as a prediction target.

Specific requirements for the report

Student will create a final electronic report (maximum 1500 written words, not including citations) using Jupyter. You are not required to create a separate pdf for it.

You must submit 1 **file:**

- A Jupyter notebook (`.ipynb` file)

Report should include the following sections:

- **Title**
- **Introduction**:
    - Clearly state the question you tried to answer with your project. Every predictive problem begins with a question that is needed to be answered.
    - Provide some relevant reasoning as to why you are asking this question. (Is this a classification or a regression problem?)
    - What is your positive label for this data (If your problem is classification, what class are you most interested in?)
- **Exploratory Data Analysis**
    - Import the data and split it into the train and test sets.
        - Read in your data.
        - Split your data into the necessary splits.
        - Summarize (describe) the data that is relevant for prediction.
    - Explain the dataset.
        - Make a note of the different features you will be using and if there is anything interesting about their behaviour.
        - Are there any null values that need to be imputed?
    - Create at least 2 visualizations that contribute to describing the data.
    - Is your data relatively balanced or will you need to do something about this?
- **Methods & Results:**

    - Describe in written English the methods you used to perform your prediction from beginning to end.
    - Your report should:
        - Create a baseline model (this will be used to compare your estimators to).
        - Identify different feature types and explain the transformations needed to apply on each feature type.
        - Transform the data (Scaling, one-hot encoding (dropping a column if binary), ordinal encoding, etc).
        - Use pipelines and column transformers when needed.
        - Test 2 different estimators using default parameters.
        - Take note that if your problem is a classification problem and your data is imbalanced , you may want to handle this by using the `class_weight` argument in your estimator.
        - Taking the better performing model from the step above, use `GridSearchCV` or `RandomizedSearchCV` to hyperparameter tune your estimator (at least 2 hyperparameters).
        - How are you scoring your model for hyperparameter tuning? Does it make sense to use "accuracy" for this data?
        - Explain which model and which hyperparameter (whether it's f1, RMSE, MAPE, recall, etc.) performed the best?
    - Using the best performing model, score your model on the test set.
    - If your problem involves classification, show the confusion matrix and classification report from your test set predictions.
    - If you used an interpretable model such as linear regression (ridge) or logistic regression, take a moment to explain any features that significantly contributed to the predictions, or a feature you were surprised did not contribute as greatly as you expected.
- **Discussion**:

    - Summarize and report your final test score along with the metric you used and compare them to the baseline model. Use the different metrics we learned to explain your results.
    - Write concluding remarks. Discuss whether this is what you expected.
    - Discuss other ideas that you did not try but could potentially improve the performance/interpretability.
    - Discuss what other questions you would like to answer.
- **References**

Notes:
- All tables and figure should have a figure/table number and a title.
- If you realize that you are repeating a lot of code try to organize it in functions. Clear presentation of your code, experiments, and results is the key to be successful in this Project. You can use code from lecture notes or assignments.

Note: You can only submit files with .ipynb extensions.

Rubric

Project Rubric			
Criteria	**Ratings**		**Pts**
Included Title and Author of the report.	**1 pts** **Submitted**	**0 pts** **No Marks** No attempt/submission	1 pts
Clearly stated the question/problem attempted to be answer.	**1 pts** **Submitted**	**0 pts** **No Marks** No attempt/submission	1 pts
Provided some relevant reasoning in why the particular question was being asked.	**1 pts** **Submitted**	**0 pts** **No Marks** No attempt/submission	1 pts

Project Rubric			
Criteria	**Ratings**		**Pts**
Described if the problem was classification or a regression problem.	**1 pts** **Submitted**	**0 pts** **No Marks** No attempt/submission	1 pts
Noted the positive label for the data.	**1 pts** **Submitted**	**0 pts** **No Marks** No attempt/submission	1 pts
Read in the data.	**1 pts** **Submitted**	**0 pts** **No Marks** No attempt/submission	1 pts
Split the data into the necessary train and test splits.	**1 pts** **Submitted**	**0 pts** **No Marks** No attempt/submission	1 pts
Used .describe() on the data and commented on interesting observations regarding the table.	**1 pts** **Submitted**	**0 pts** **No Marks** No attempt/submission	1 pts
Included details about different features being used, and explained why certain columns may or may not have been dropped/included.	**1 pts** **Submitted**	**0 pts** **No Marks** No attempt/submission	1 pts

Project Rubric			
Criteria	**Ratings**		**Pts**
Included .info() and explained if there were any null values in the dataset and the course of action planned to remove them (such as using SimpleImputer).	**1 pts** **Submitted**	**0 pts** **No Marks** No attempt/submission	1 pts
Created one visualization regarding a feature or statistic from the data.	**1 pts** **Submitted**	**0 pts** **No Marks** No attempt/submission	1 pts
Created a second or supporting visualization describing the data and its features.	**1 pts** **Submitted**	**0 pts** **No Marks** No attempt/submission	1 pts
If the problem was regarding classification - explained the distribution of the target column and if the classes are balanced.	**1 pts** **Submitted**	**0 pts** **No Marks** No attempt/submission	1 pts
Created a DummyClassifier or DummyRegressor.	**1 pts** **Submitted**	**0 pts** **No Marks** No attempt/submission	1 pts
Specified the columns being used for the feature table (X).	**2 pts** **Submitted**	**0 pts** **No Marks** No attempt/submission	2 pts

Project Rubric			
Criteria	**Ratings**		**Pts**
Explained if the features were binary, categorical, numeric.	**2 pts** **Submitted**	**0 pts** **No Marks** No attempt/submission	2 pts
Explained the reasoning behind any dropped feature.	**2 pts** **Submitted**	**0 pts** **No Marks** No attempt/submission	2 pts
Explained the transformations needed to apply on each feature type. (StandardScaler, SimpleImputer, CountVectorizer, OneHotEncoder).	**1 pts** **Submitted**	**0 pts** **No Marks** No attempt/submission	1 pts
Transformed the columns appropriately (scaling, one-hot encoding, ordinal encoding, etc).	**1 pts** **Submitted**	**0 pts** **No Marks** No attempt/submission	1 pts
Used Pipeline or make_pipeline to transform the columns of each feature type.	**1 pts** **Submitted**	**0 pts** **No Marks** No attempt/submission	1 pts
Used ColumnTransformer or make_column_transformer to transform the columns together.	**1 pts** **Submitted**	**0 pts** **No Marks** No attempt/submission	1 pts

Project Rubric			
Criteria	**Ratings**		**Pts**
Used class_weight=balanced for the estimator.	**1 pts** **Submitted**	**0 pts** **No Marks** No attempt/submission	1 pts
Explained the reasoning of why a specific model is being used (SVC, SVM, KNN, Decision Tree, Linear/logistic Regression).	**1 pts** **Submitted**	**0 pts** **No Marks** No attempt/submission	1 pts
Tuned 1 Hyperparameter using GridSearchCV or RandomizedSearchCV.	**1 pts** **Submitted**	**0 pts** **No Marks** No attempt/submission	1 pts
Specified why they chose GridSearchCV vs RandomizedSearchCV.	**1 pts** **Submitted**	**0 pts** **No Marks** No attempt/submission	1 pts
Tuned a second Hyperparameter.	**1 pts** **Submitted**	**0 pts** **No Marks** No attempt/submission	1 pts
Scored the model appropriately explaining the choice of metric used for hyperparameter tuning.	**1 pts** **Submitted**	**0 pts** **No Marks** No attempt/submission	1 pts
Explained which model and which hyperparameters (whether it's f1, RMSE,	**1 pts** **Submitted**	**0 pts** **No Marks** No attempt/submission	1 pts

Project Rubric			
Criteria	**Ratings**		**Pts**
MAPE, recall, etc.) performed the best.			
Reported the best validation score for the model with the best hyperparameters.	**1 pts** **Submitted**	**0 pts** **No Marks** No attempt/submission	1 pts
Scored the best performing model on the test set.	**1 pts** **Submitted**	**0 pts** **No Marks** No attempt/submission	1 pts
If the problem involves classification; showed the confusion matrix from the test set predictions.	**1 pts** **Submitted**	**0 pts** **No Marks** No attempt/submission	1 pts
If the problem involves classification; show the classification report from the test set predictions.	**1 pts** **Submitted**	**0 pts** **No Marks** No attempt/submission	1 pts
If an interpretable model such as linear regression (ridge) or logistic regression was used, interpreted the model's coefficients.	**1 pts** **Submitted**	**0 pts** **No Marks** No attempt/submission	1 pts
Summarized and reported the final test scores and metrics.	**1 pts** **Submitted**	**0 pts** **No Marks** No attempt/submission	1 pts

Project Rubric			
Criteria	**Ratings**		**Pts**
Compared the model's results to the baseline model.	**1 pts** **Submitted**	**0 pts** **No Marks** No attempt/submission	1 pts
Used multiple different metrics to explain the results.	**1 pts** **Submitted**	**0 pts** **No Marks** No attempt/submission	1 pts
Wrote concluding remarks.	**1 pts** **Submitted**	**0 pts** **No Marks** No attempt/submission	1 pts
Discussed other ideas that could be done to improve the performance/interpretability.	**2 pts** **Submitted**	**0 pts** **No Marks** No attempt/submission	2 pts
Citations	**1 pts** **Submitted**	**0 pts** **No Marks** No attempt/submission	1 pts
Flawless spelling.	**2 pts** **Excellent** No mistakes	**1 pts** **Good** Less than 15 spelling mistakes.	**0 pts** **No Marks** No attempt/submission	2 pts

Project Rubric				
Criteria	**Ratings**			**Pts**
No grammar mistakes.	**2 pts** **Excellent** No mistakes	**1 pts** **Good** Less than 15 spelling mistakes.	**0 pts** **No Marks** No attempt/submission	2 pts
The submission is around 1500 words - not including code or citations	**1 pts** **Submitted**	**0 pts** **No Marks** No attempt/submission		1 pts
Used functions where applicable and adhered to the DRY principle	**1 pts** **Submitted**	**0 pts** **No Marks** No attempt/submission		1 pts
Code is well thought out and organized. Code is human readable with appropriate comments and self-explanatory variable names.	**1 pts** **Submitted**	**0 pts** **No Marks** No attempt/submission		1 pts
Group work → Equal distribution of work, participation, feedback of individual group member				20 pts
Bonus	**You tried 4 or more algorithms to get good results – 10 pts.** **Apply the under sampling or over sampling technique to get the better results – 5 pts.**			15 pts

Project Rubric		
Criteria	**Ratings**	**Pts**
Total Points: 70		

If student break the golden rule at any point in the project whole project will be marked as 0.