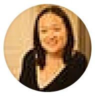
Exploring machine learning models to predict dementia



Stephanie Tsao

18 min read · Dec 10, 2023

By University of Michigan researchers Pedro Hermon, Jonathan, Yoon, and Stephanie Tsao

Table of Contents

Research suggests that roughly one in three Americans age 65 years and older have had dementia or mild cognitive impairment (Manly et al.). Dementia, a condition characterized by cognitive decline and memory loss, creates significant costs if patients require long-term care. As the U.S. population ages, our team explored if we can use data to develop a machine learning model that predicts positive cases of dementia. Such a model can be a tool for insurance companies to evaluate applicants seeking long-term care (LTC) insurance and to accelerate the underwriting process.

In this blog, we describe our methods, results, the ethical considerations, and potential social biases associated with using machine learning models in the context of dementia and LTC insurance underwriting. We intend to foster a conversation to explore the challenges and responsibilities of leveraging technology like machine learning models for decision-making in sensitive domains.

Problem statement

We would like to build a machine learning model to answer the following questions:

- Can machine learning models help predict the likelihood of dementia in various tissue samples?

- Can we use machine learning to identify classes of genes that are associated or correlated with dementia?

We envisioned a successful outcome to have the following facets:

- Stable predictions that predict the Dementia group with a high level of precision
- Model predictions that indicate similar genes or groups of genes are affiliated with dementia that can be corroborated through external research

Broader impacts

A predictive model for dementia can be another tool for insurance companies to assess an applicant's health in an efficient manner. Some of our team members have applied for LTC insurance and experienced an underwriting process that can span several months. The lengthy time can create costs and unnecessary stress for patients, particularly if they cannot receive coverage for LTC by other means. LTC insurance provides financial assistance for individuals who need extended care or assistance with activities of daily living due to chronic illnesses, disabilities, or cognitive impairments. Because this care may not be considered "medical care," LTC is typically not covered by health insurance or federal health insurance programs like Medicare (U.S. Centers for Medicare and Medicaid Services).

This circumstance emphasizes the critical need to improve the speed and quality of the underwriting process. Our exploration into the integration of machine learning in this context is about accurately predicting positive cases and expediting a process that can significantly impact the lives of individuals and families. By streamlining the underwriting process, we aim to contribute to a more timely and effective process for obtaining LTC coverage.

We are aware that machine learning models can have broader impacts that include unintended negative consequences. For instance, the brain tissue data that we relied on to build our model came from deceased donors who identified mostly as white (Figure 1).

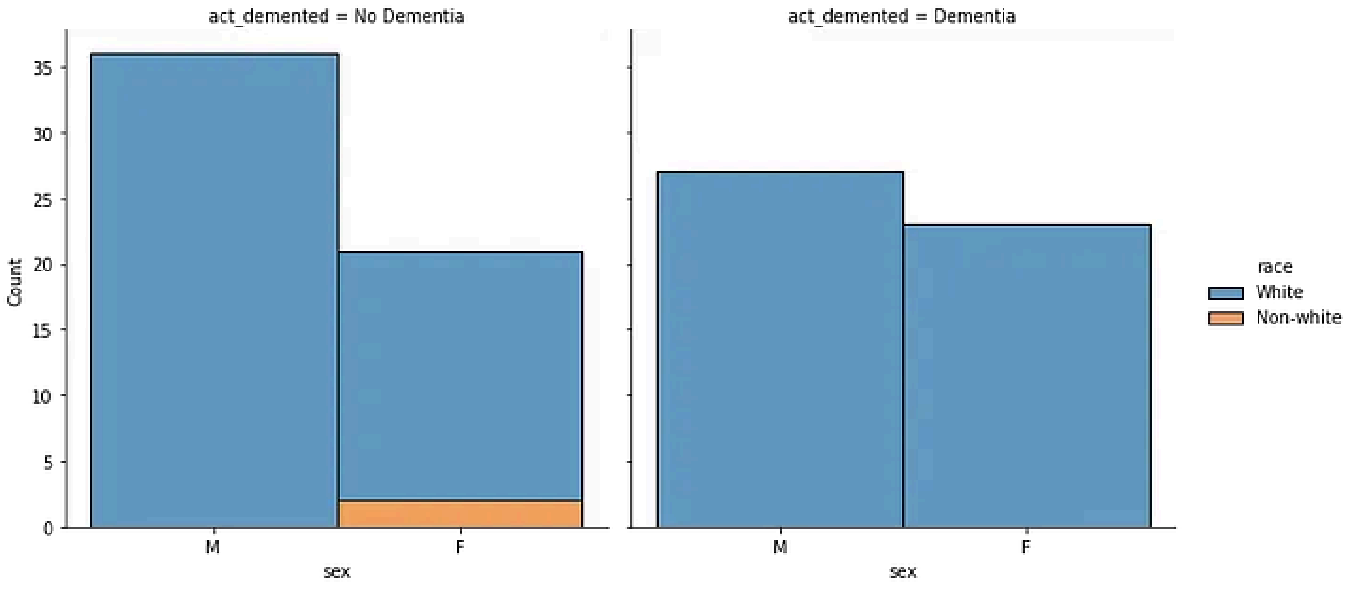


Figure 1. Number of donors without dementia (left) and with dementia (right) by gender and race

When designing such models, researchers have to be aware about what types of people are unaccounted for and could be inaccurately assessed by their model. Prior research shows that dementia is more prevalent in non-Hispanic Black individuals, who have limited representation in our dataset (Manly et al.). This lack of representation can lead to a model that can inaccurately predict dementia in non-white individuals.

Although age is the most “potent” risk factor for dementia, our team did not prefer to use age as a predictor because not all ages were represented in our dataset (Manly et al.). Prior studies include patients at least 65 years old in dementia research (Javeed et al.). Because our dataset comes from donors 77 years old to 100 years old, our model may inaccurately predict dementia cases with younger age groups who are in their 60s to 70s (Figure 2).

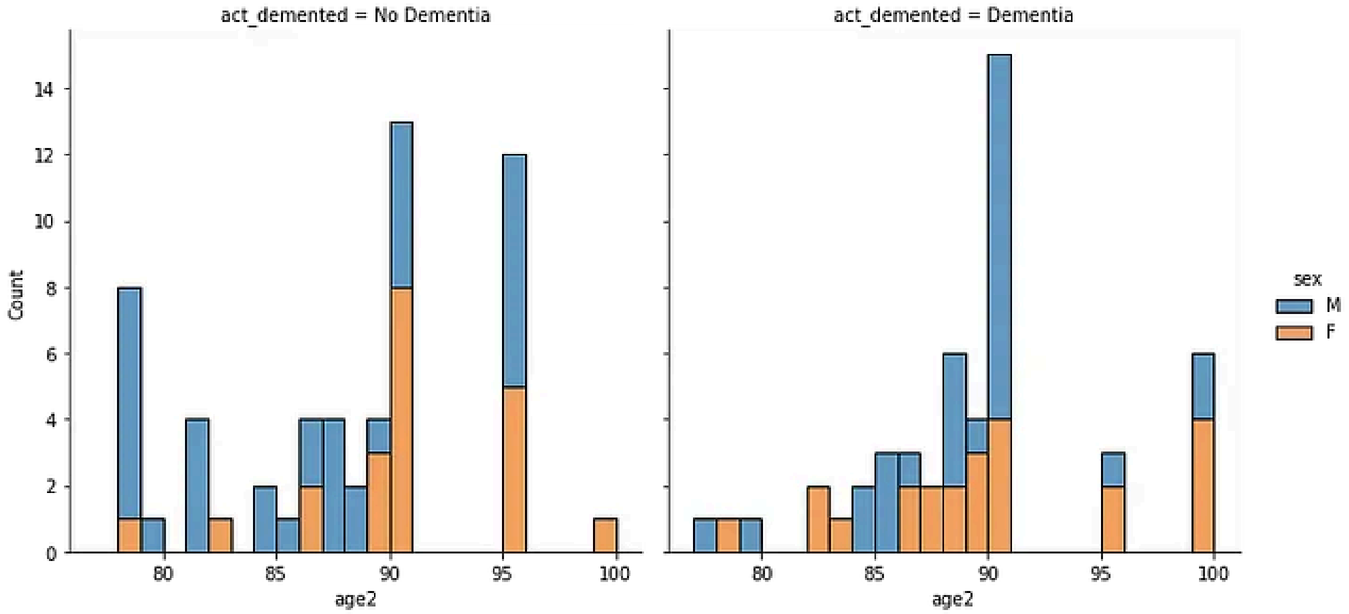


Figure 2. Number of donors without dementia (left) and with dementia (right) by age and gender

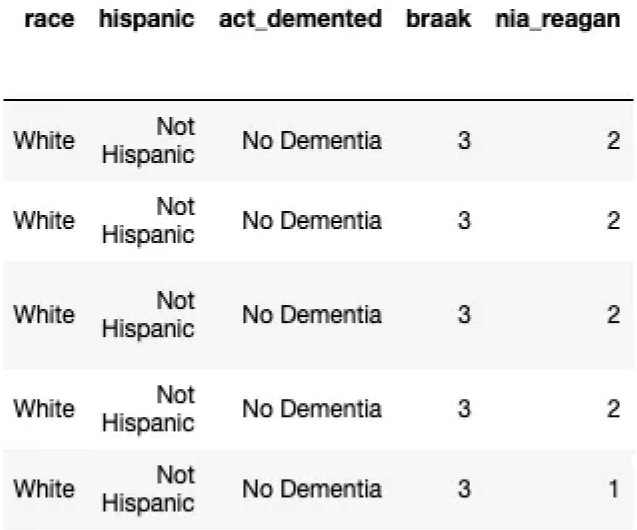
We discuss possible ways of reducing these unintended consequences in the Future Work section.

Methodology

We downloaded brain tissue data from the Aging, Dementia, and Traumatic Brain Injury Project. We provide more information about how we use Python-based libraries to analyze the data in our Github repository.

Our dataset is both small and highly dimensional, meaning it has many columns. We have brain tissue data from 110 donors, but the genetic information from each tissue sample produced over 50,000 columns.

The *DonorInformation.csv* file has a column named *act_demented* that labels each tissue sample as either “Dementia” or “No Dementia.” Using this column, we created supervised learning models to predict these labels.



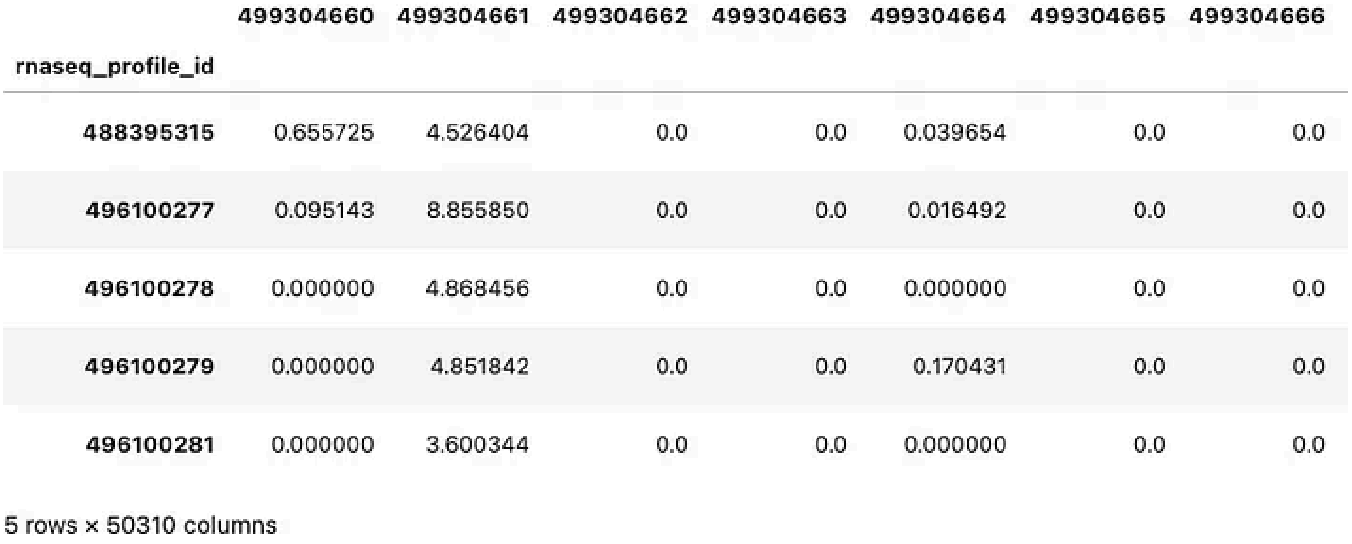
race	hispanic	act_demented	braak	nia_reagan
White	Not Hispanic	No Dementia	3	2
White	Not Hispanic	No Dementia	3	2
White	Not Hispanic	No Dementia	3	2
White	Not Hispanic	No Dementia	3	2
White	Not Hispanic	No Dementia	3	1

Table 1. Select columns from DonorInformation.csv file including “act_demented” column used for model predictions

Data cleaning and merging

To make predictions, we combined genetic data from the *fpkm_table_normalized.csv* file with the *act_demented* column from the *DonorInformation.csv*. Our primary dataframe included a row for each *rna_profile_id* and a column for each *gene_id*. If we think of expression rates as indicators of how much that gene is turned on or off, the dataframe provides the level each gene is turned on or off within each tissue sample. Each sample is run through a technique called RNA sequencing to produce the gene information.

A snapshot of select columns in our primary dataframe is below:



	499304660	499304661	499304662	499304663	499304664	499304665	499304666
rnaseq_profile_id							
488395315	0.655725	4.526404	0.0	0.0	0.039654	0.0	0.0
496100277	0.095143	8.855850	0.0	0.0	0.016492	0.0	0.0
496100278	0.000000	4.868456	0.0	0.0	0.000000	0.0	0.0
496100279	0.000000	4.851842	0.0	0.0	0.170431	0.0	0.0
496100281	0.000000	3.600344	0.0	0.0	0.000000	0.0	0.0

5 rows × 50310 columns

Table 2. Select columns of primary dataframe

We sourced RNA expression rates from the normalized file over the *fpkm_table_unnormalized.csv* file because this data normalizes the expression rates by the length of the gene, which allows for more accurate comparability across genes.

Model selection

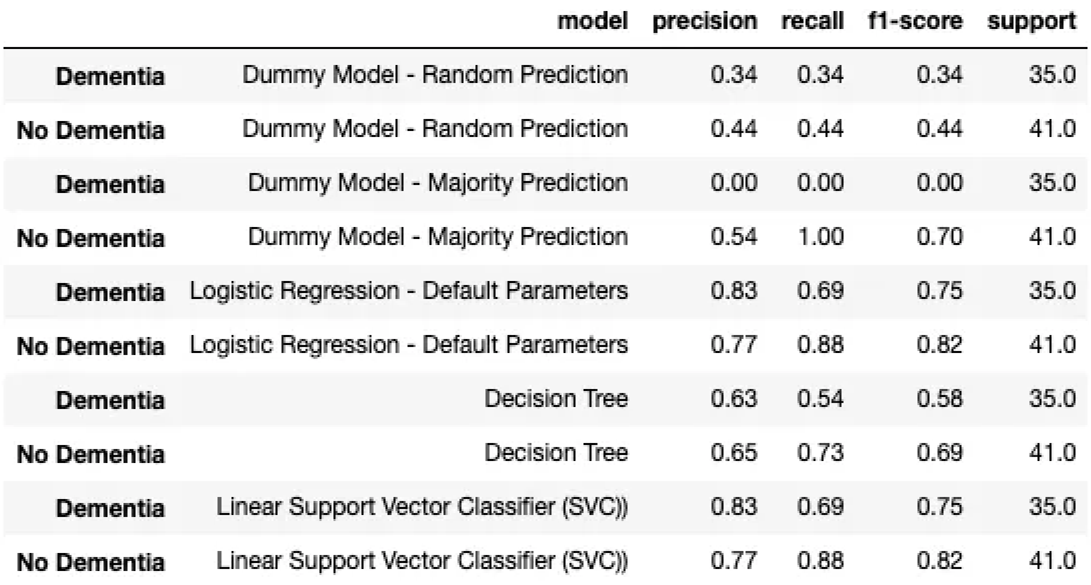
In the first phase of our modeling, we incorporated all gene columns. We tested five models. Because our goal was to create a model that predicted two classes: dementia or no dementia, we tested two common classification models that can handle high dimensional datasets: a logistic regression and linear support vector classifier (SVC).

A third, more complex model we tested was a decision tree that uses a series of if/then questions to decide if a tissue sample is indicative of dementia or not (Müller and Guido). The tree approach builds a structure like a family tree where the model uses layers of questions to arrive at a decision. We opted for a decision tree model because it gives us the flexibility to control the depth, or how deep the model learns to prevent overfitting, a situation when a model learns from a dataset well but predicts poorly off new and unseen data. The risk of overfitting means our model will not generalize or apply well to other tissue samples. We also chose a decision tree because it was among a few models that performed well in predicting Alzheimer's Disease, a common cause of dementia, when trained on electronic health records (Javeed et al.).

Lastly, we created two dummy models to serve as a baseline for comparison. The first dummy model randomly predicts classes based on the distribution of the target variable. The second dummy model always predicts the majority class. These dummy models help establish whether our best model performs better than a simple model that predicts at random and a second dummy model that just predicts the more common class.

Model performance and evaluation

Training and testing these models revealed the logistic regression and SVC models performed the best among our group.



	model	precision	recall	f1-score	support
Dementia	Dummy Model - Random Prediction	0.34	0.34	0.34	35.0
No Dementia	Dummy Model - Random Prediction	0.44	0.44	0.44	41.0
Dementia	Dummy Model - Majority Prediction	0.00	0.00	0.00	35.0
No Dementia	Dummy Model - Majority Prediction	0.54	1.00	0.70	41.0
Dementia	Logistic Regression - Default Parameters	0.83	0.69	0.75	35.0
No Dementia	Logistic Regression - Default Parameters	0.77	0.88	0.82	41.0
Dementia	Decision Tree	0.63	0.54	0.58	35.0
No Dementia	Decision Tree	0.65	0.73	0.69	41.0
Dementia	Linear Support Vector Classifier (SVC))	0.83	0.69	0.75	35.0
No Dementia	Linear Support Vector Classifier (SVC))	0.77	0.88	0.82	41.0

Table 3. Performance metrics for phase one models

Although we compiled various performance measures including precision, recall, and f1-score, we paid more attention to precision. Our goal was to predict positive cases of dementia and reduce false positive cases, instances where our model misclassified a sample as having dementia when it actually did not. A false positive classification can have the unintended consequence of an insurance company setting a long-term care insurance applicant's price too high.

The logistic regression and SVC each predicted positive cases of dementia with a precision of 83%, compared to 63% for the decision tree model and 34% for the dummy model predicting randomly. Decision trees, while effective, may be more prone to overfitting and might not generalize as well as logistic regression on unseen data. The logistic regression and SVC models also achieved higher f1-scores, which suggests they effectively captured the underlying patterns in the data and are both strong candidates for our best performing model.

Another evaluation metric we employed were confusion matrices, which compare the number of actual cases for "Dementia" and "No Dementia" against our predictions. The confusion matrix presents a count of how many cases were predicted accurately for the dementia cases (in the top row) and the "No Dementia" cases in the bottom row. For instance, the logistic regression model predicted 24 tissue samples as having dementia that actually did, but 11 samples were falsely labeled with dementia (Figure 3). Similarly, the logistic regression model predicted 36 tissue samples correctly as not having dementia, but misclassified 5 samples.

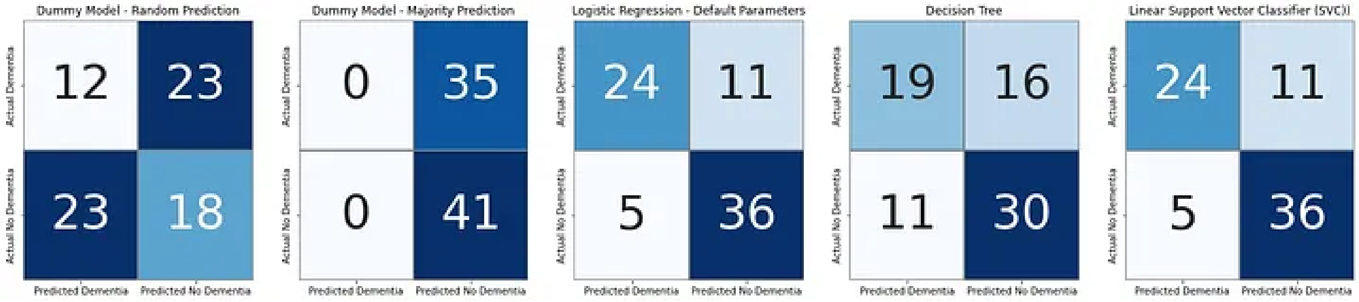


Figure 3. Model results shown in confusion matrices

A third evaluation method we employed is examining the receiver operating characteristics or ROC curve, which compares how well our model predicts actual cases of dementia versus false positive cases. In Figure 4, we see that the model has good discriminatory ability based on the area under the curve (AUC). The more the ROC curves to the upper left, the more accurate (Nahm, FS). The curve is inclined towards the top-left corner, which indicates a good balance between true positive rate and false positive rate. You can also see that the model performs better than random guessing (AUC = 0.5).

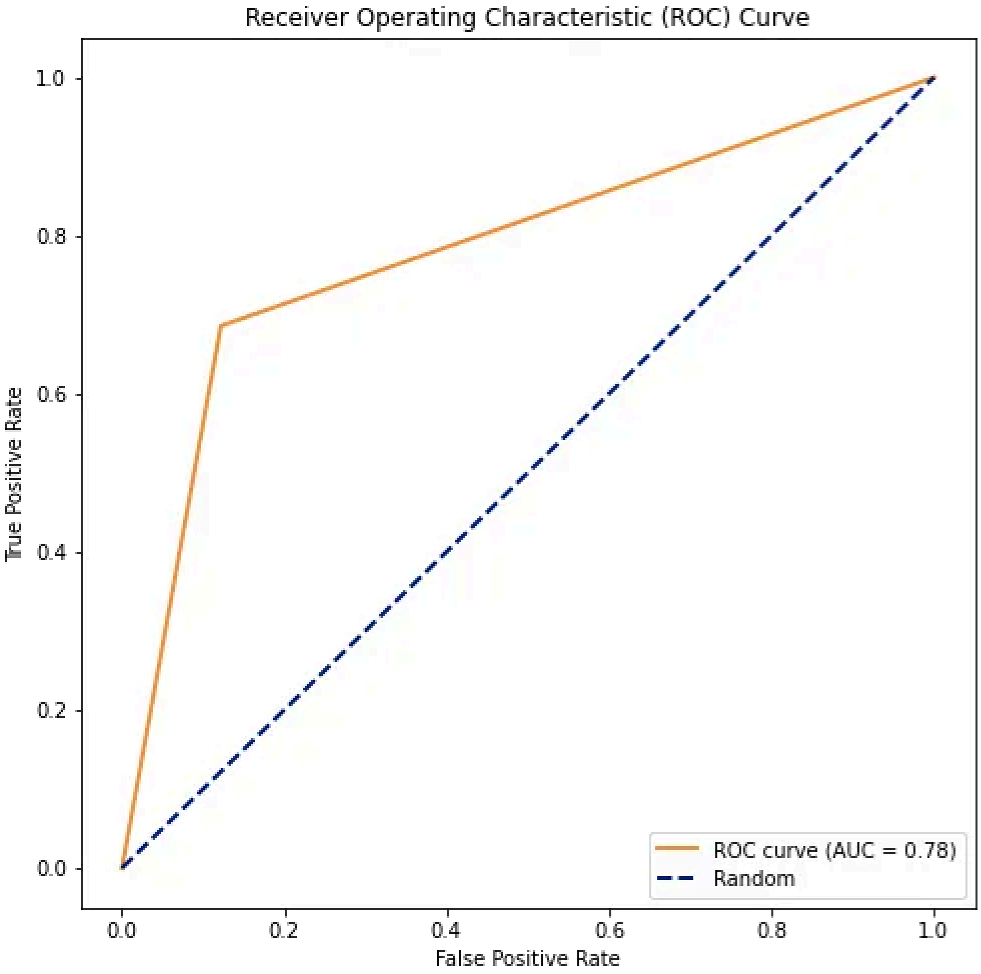


Figure 4. ROC curve

Because the SVC and logistic regression models had similar performance, we chose the logistic regression as our best model because of our team's familiarity with this model's parameters. We decided to set aside the linear SVC as a model that we can test in the future because we are aware that these models may be computationally more efficient depending on the data and the decision boundaries (Zazo and Protopapas).

Model tuning and further evaluation

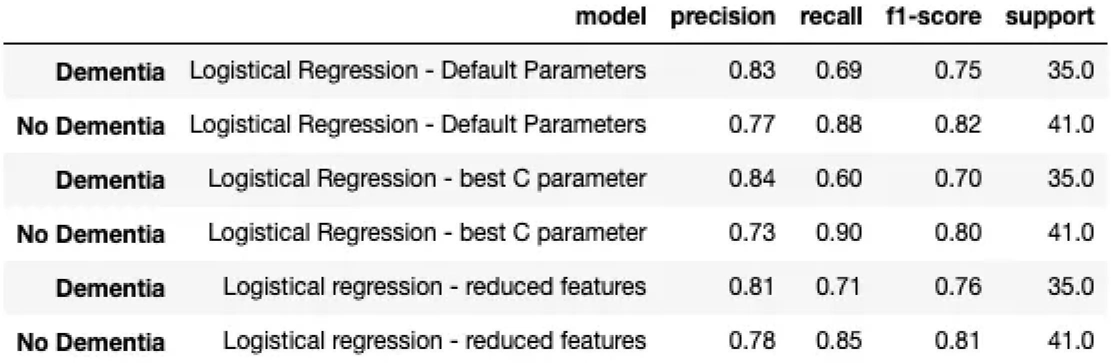
We explored adjusting hyperparameters to see their effect on how our best model runs. For instance, we adjusted the constant C, which affects how strongly the model is biased to our training data. We explored using various values for C: 0.001, 0.01, 0.1, 1, 10, 100, and 1,000. The higher the C parameter, the more the model learns our training dataset well but will predict poorly with other datasets (Muller and Guido 57–58). Using sklearn's GridSearchCV feature, we ran our logistic regression model with each of the C values and found that the lowest value, 0.001, achieved the highest accuracy score at 76%.

```
from sklearn.model_selection import GridSearchCV
# Define the logistic regression model
logreg_model = LogisticRegression()

# Define the hyperparameter grid for C
param_grid = {'C': [0.001, 0.01, 0.1, 1, 10, 100, 1000]}

# Perform grid search with 5-fold cross-validation
grid_search = GridSearchCV(logreg_model, param_grid, cv=5, scoring='precision')
grid_search.fit(X_train_scaled, y_train)

# Print the best hyperparameter value
best_C = grid_search.best_params_['C']
print(f"Best C value: {best_C}")

# Evaluate the model with the best hyperparameter on the test set
best_model = grid_search.best_estimator_
test_accuracy = best_model.score(X_test_scaled, y_test)
print(f"Accuracy on test set with best C: {test_accuracy}")
```

```
Best C value: 0.001
Accuracy on test set with best C: 0.7631578947368421
```

We also tried different solvers, which provides the optimization algorithm used in our model. We ran our logistic model with the C = 0.001 and replaced the solver by trying liblinear, newton-cg, lbfgs, sag, and saga. Ultimately we found that liblinear, newton-cg, sag, and saga solvers achieved a cross validation score in the range of 0.51–0.55, while the default solver of lbfgs achieved the highest score of 0.63. Future iterations of this work can test our best C value with the lbfgs solver and the L1 regularization penalty, which helps regulate the amount of features used to make predictions.

In the following table, we noticed that the precision increased slightly to 84% from 83% for the dementia class when incorporating the best C value of 0.001. It is crucial to observe that this uptick in precision comes with a marginal decrease in recall, another metric that accounts for dementia cases that are falsely predicted as negative. This reduction is attributed to the conservative nature of the model, which aims to minimize false positives, potentially resulting in the omission of some true positives.



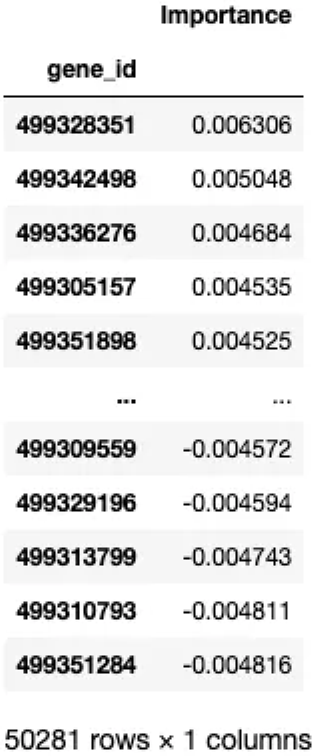
	model	precision	recall	f1-score	support
Dementia	Logistical Regression - Default Parameters	0.83	0.69	0.75	35.0
No Dementia	Logistical Regression - Default Parameters	0.77	0.88	0.82	41.0
Dementia	Logistical Regression - best C parameter	0.84	0.60	0.70	35.0
No Dementia	Logistical Regression - best C parameter	0.73	0.90	0.80	41.0
Dementia	Logistical regression - reduced features	0.81	0.71	0.76	35.0
No Dementia	Logistical regression - reduced features	0.78	0.85	0.81	41.0

Table 4. Performance metrics for adjusted logistic regression model

We also explored different tools to filter down the more than 50,000 columns, but were surprised that our attempts to filter features reduced the precision of the Dementia class to 81% from 83% in the original logistic regression model without any tuned parameters. In the next section, we will cover more about how we tried to filter features.

Feature selection

Two tools that helped us filter features is sklearn's "feature importance" function and recursive feature elimination, or RFE. First we used the "feature importance" method to sort the genes from highest to lowest in terms of their importance to our predictions. A subset of the sorted list is provided.



	Importance
gene_id	
499328351	0.006306
499342498	0.005048
499336276	0.004684
499305157	0.004535
499351898	0.004525
...	...
499309559	-0.004572
499329196	-0.004594
499313799	-0.004743
499310793	-0.004811
499351284	-0.004816

50281 rows × 1 columns

Table 5. Output of feature importance

A histogram helped us see that there are over 14,000 genes that have an importance value of 0, which indicates that we can remove these features if they do not add any advantage to our model (Figure 5). The challenge we faced was figuring out how many features to omit.

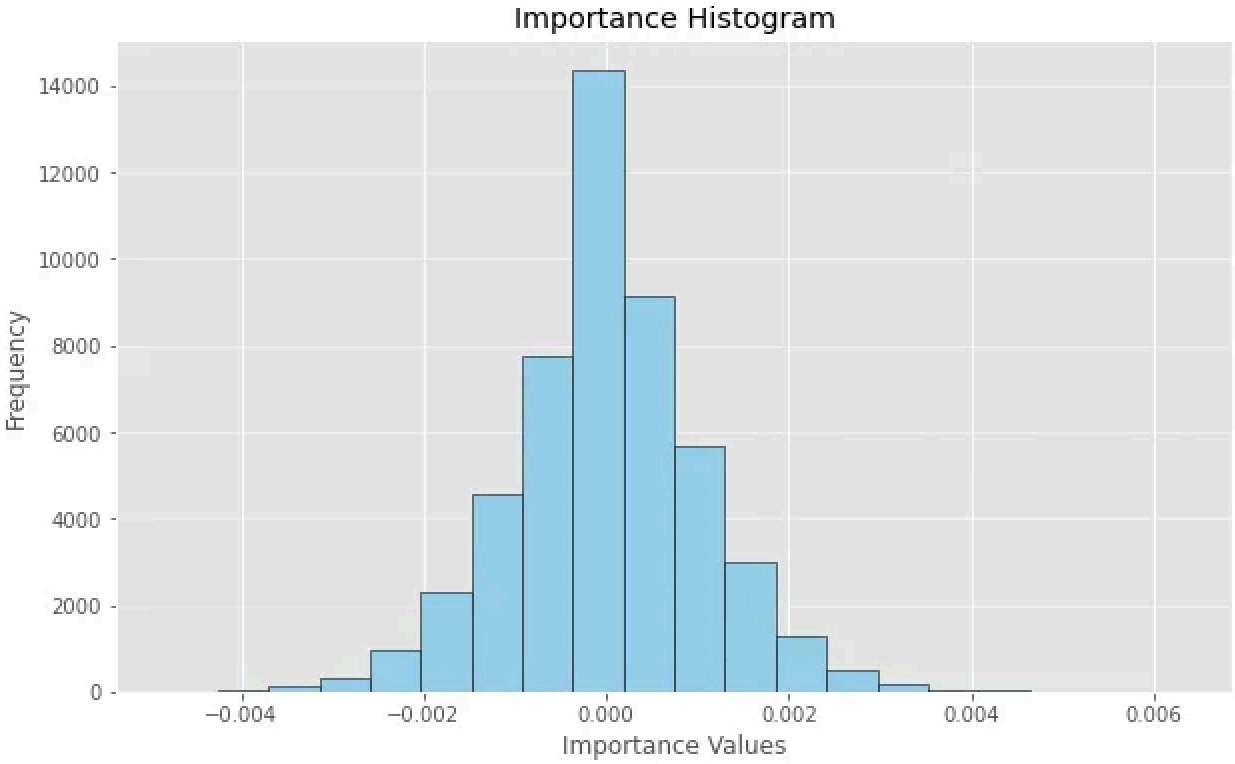


Figure 5. Histogram of feature importance values

We also used recursive feature elimination (RFE) to find an optimal number of features keep. This approach runs our best logistic regression model on a max number of features and then reruns the model after discarding unimportant features. Because the RFE approach was computationally intensive when using all 50,281 features, we tried *max_features* values of 300; 500; 1,000; 5,000; and 10,000.

We found that setting *max_features* to 5,000 was the most optimal because this produced the most number of genes that have been validated through external research as contributing to dementia. In Figure 6, we marked a red line at 3,622 features, where the RFE showed the highest accuracy when we limited *max_features* to 5,000.

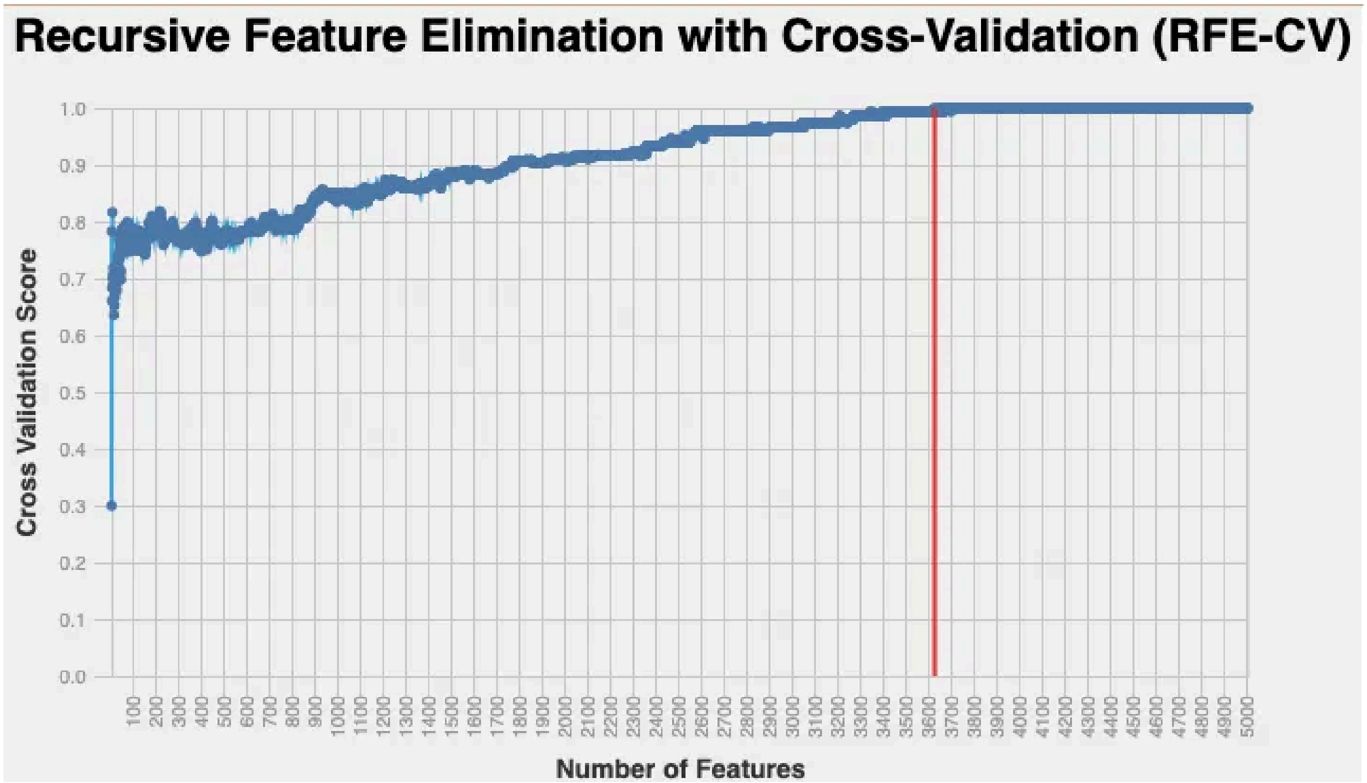
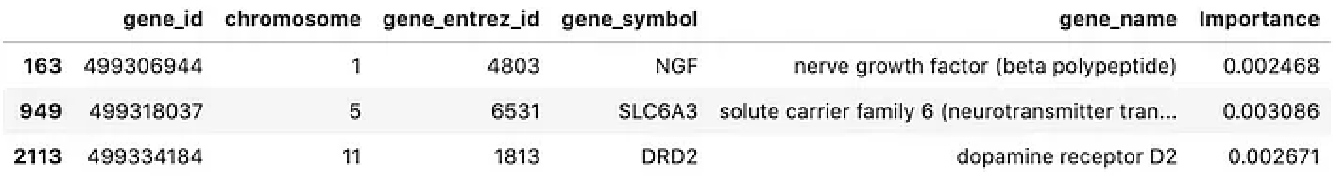


Figure 6. Recursive feature elimination shows the optimal features achieving the highest accuracy score

We compared the features selected through the RFE approach against a list of 121 genes associated with dementia, which we obtained from an external database called MalaCard (Rappaport). We saved this list in a file called *dementia_associated_genes.csv*. A comparison between these 121 genes and the genes from our RFE analysis revealed only three genes in common: NGF, SLC6A3, and DRD2.



	gene_id	chromosome	gene_entrez_id	gene_symbol	gene_name	importance
163	499306944	1	4803	NGF	nerve growth factor (beta polypeptide)	0.002468
949	499318037	5	6531	SLC6A3	solute carrier family 6 (neurotransmitter tran...	0.003086
2113	499334184	11	1813	DRD2	dopamine receptor D2	0.002671

Table 6. Three genes in our input features that had associations with dementia based on third-party research

When we adjusted *max_features* to 10,000, the RFE approach took more computing power without producing more genes in common with the MalaCard list. Setting *max_features* at 1,000 or less showed no genes in common; thus, we set with *max_features* at 5,000 to produce a list of the 3,622 most important features.

When running our best logistic regression model with the 3,622 features, we surprisingly saw slightly lower precision in predicting Dementia cases. The precision fell to 81% compared to 83% to 84% in the logistic regression with default parameters and tuned model, respectively. Predictions of No Dementia cases were slightly better than our earlier models. Our new model predicted No Dementia cases with 78% precision, compared to 77% in a logistic regression with no hyperparameter tuning and 73% precision in a model with the best C value.

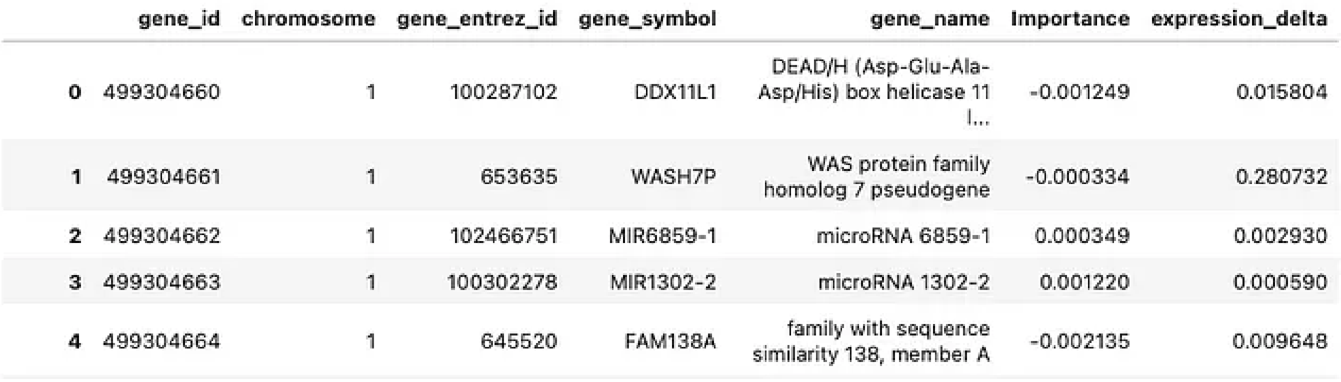
Despite employing only 3,622 features or the representation of 3,622 genes, there was little variation in precision levels for either class. This suggests that reducing dimensionality can have similar performance with faster model performance than using all 50,281 features.

Further gene analysis

Reducing the number of genes did not reveal significant improvement in our model predictions, so we shifted to produce a visual to understand the type of genes were are using as inputs. Specifically, we wanted to answer the question: would limiting our inputs to genes similar to the three that MalaCard shows as associated with dementia improve our model performance?

Although we did not get far with this part of our research, we used a visual to help us understand which genes might be connected to the 3 genes in Table 6.

To supplement our analysis, we used natural language processing, specifically sklearn's *tfidfvectorizer*, to process the gene descriptions and identify related genes or genes with similar descriptions. Additionally, we sought to visualize changes in the gene expression levels between non-dementia and dementia patients. Below is a sample of the dataframe.



	gene_id	chromosome	gene_entrez_id	gene_symbol	gene_name	Importance	expression_delta
0	499304660	1	100287102	DDX11L1	DEAD/H (Asp-Glu-Ala-Asp/His) box helicase 11 l...	-0.001249	0.015804
1	499304661	1	653635	WASH7P	WAS protein family homolog 7 pseudogene	-0.000334	0.280732
2	499304662	1	102466751	MIR6859-1	microRNA 6859-1	0.000349	0.002930
3	499304663	1	100302278	MIR1302-2	microRNA 1302-2	0.001220	0.000590
4	499304664	1	645520	FAM138A	family with sequence similarity 138, member A	-0.002135	0.009648

Table 7. Sample dataframe

This holistic approach enhances our model's interpretability and may provide insights into the potential biological mechanisms underlying dementia onset.

With this supplemental analysis, we visualized related genes using the NetworkX library. In Figure 7, the objective was to condense the significant features identified by the model. The gene description was processed using a *tfidf_vectorizer* and *cosine similarity* was used as a measure of similarity to quantify the relationships among the features. This method facilitates a comprehensive understanding of the interplay between crucial gene-related characteristics, enhancing the interpretability of the model's insights.

Each dot represents a gene, and the distance reflects how strongly they are connected to a target, which we set as the gene NGF, one of the three genes noted in Table 6. The dots are sized by how differently they were expressed in Dementia and No Dementia cases. A larger difference might suggest a correlation with dementia and a feature worth further testing in our model.

Because we were limited on time, we envisioned a future step as focusing on those genes that are close to and connected to the NGF dot. We can research what those genes are and possibly test our model with that subset. Removing those genes on the outer rim of the network diagram can offer another approach to help us reduce our inputs to noteworthy genes. In the future, we would like to test our models with those genes in the center of the network to see if genes correlated with dementia impact model performance.

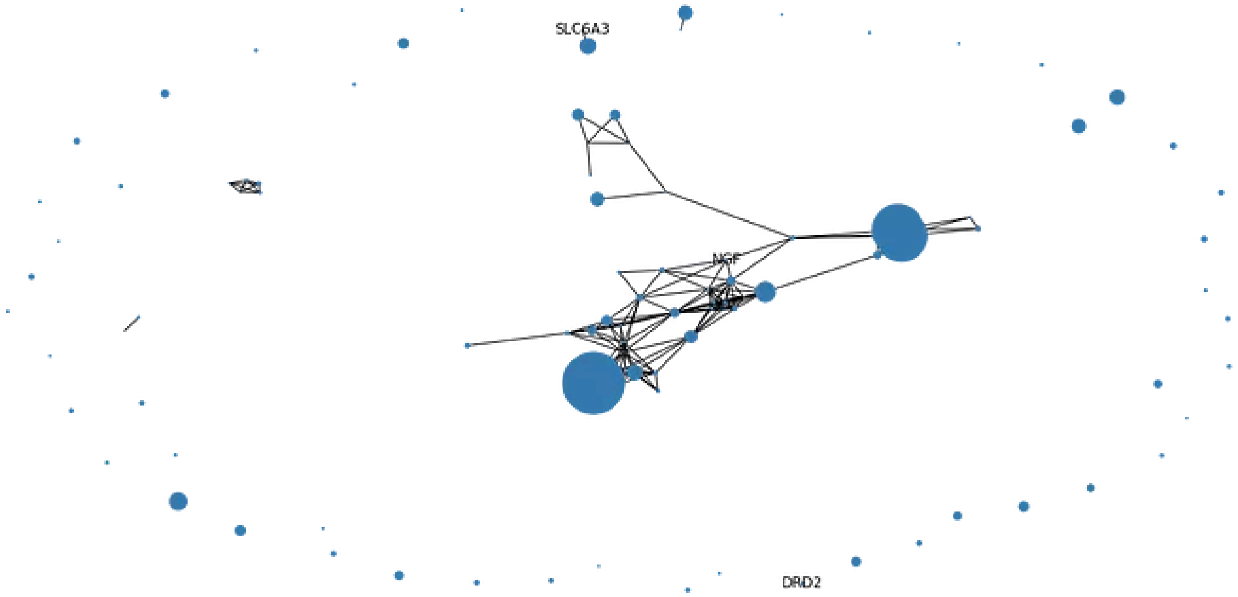


Figure 7. NetworkX diagram of relations to target gene NGF

Model interpretability and instructor feedback

We knew our small dataset and the fact that all donors came from a single state in the U.S. raised concerns about potential overfitting. We received questions about how our model would predict for donors from other states or countries. Our project coach suggested that we monitor for any flags in our data that signal possible overfitting. With this feedback, we assessed the model's performance using cross-validation, a statistical method that trains a model on multiple subsets of the data.

We also checked the training and test performance for signs of overfitting. Our findings reveal that while training accuracies are generally high, which suggests effective learning from the available data, there are noticeable variability in accuracy levels across different folds (Figure 8). Notably, test accuracies are lower than training accuracies, as anticipated, but the disparities among folds hint at the possibility of overfitting. In the Future Work section, we discuss the possible consequences of overfitting and ways we can adjust our research methods to prevent overfitting.

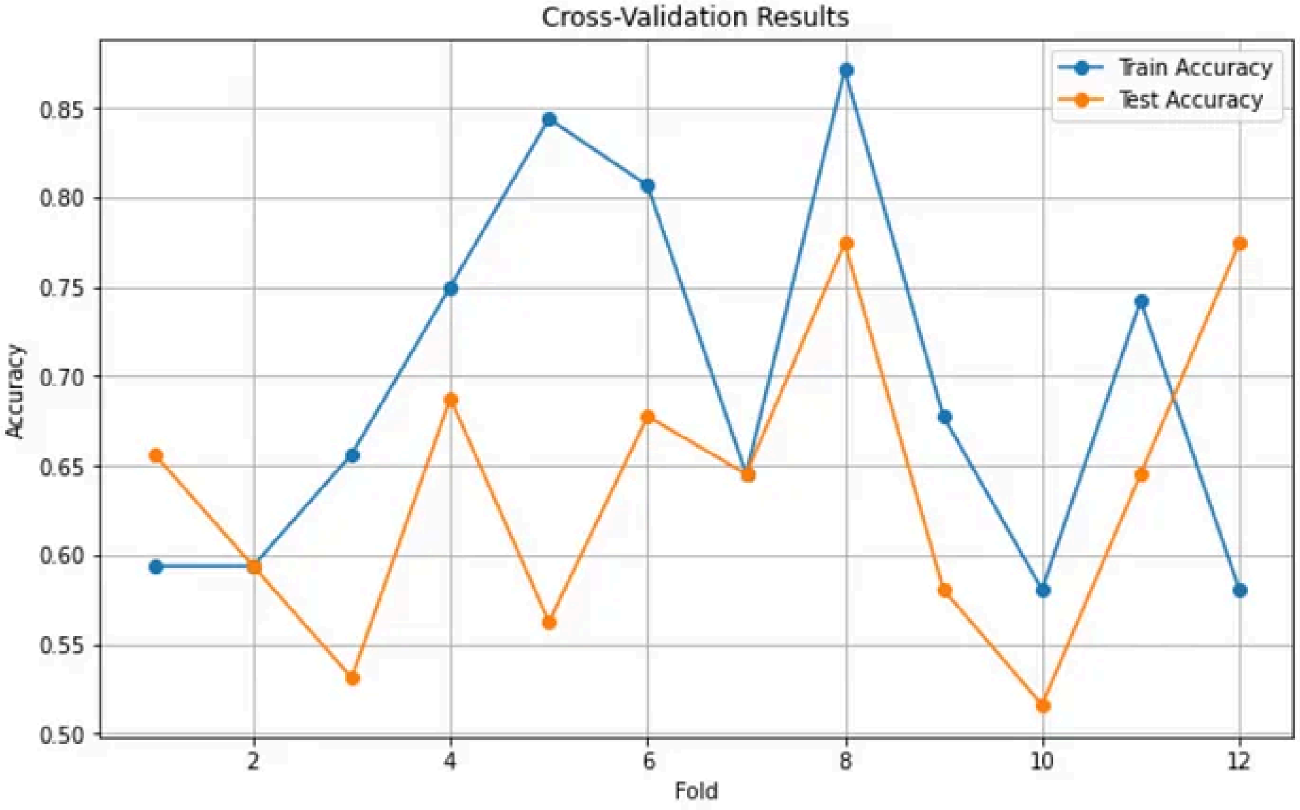


Figure 8. Comparison of train and test accuracy

Future work

We envision a next stage of this research to focus on various topics. First, future work can look at how to increase the generalizability of the model. At the start of this project, we received questions from our peers and course instructors about how our model can predict dementia in living patients if it learns from tissue samples from deceased donors. Getting brain tissue samples from living patients would be invasive; thus, a next stage of this research could test our model on tissue samples from other tissue types such as saliva samples or blood tests, which are less invasive methods of obtaining genetic material.

Another area of future work is to build a model using a larger sample of donors to reduce the risk of bias. We found another dataset with 230 brain tissue samples from autopsies but were not able to merge the dataset into because of incompatible formats (Zhang et al.). In the future, we can also consider building a model based on other data types such as MRI scans. Feeding images into a model to predict dementia provides the option of using images of living patients. Prior research has shown that predictive models trained on image data, such as MRI brain scans, predicted Alzheimer’s Disease better than models using clinical data or vocal recordings of patients or a mix of both (Javeed et al.).

Although we did not use the number of traumatic brain injuries (TBI) column in our dataset, we noticed that more men than women suffered up to three TBI (Figure 9). Research shows that more TBIs can increase one’s risk of dementia (Mendez). Another area of future work is to control for the level of TBIs in our model or factor this in as a feature in our prediction model.

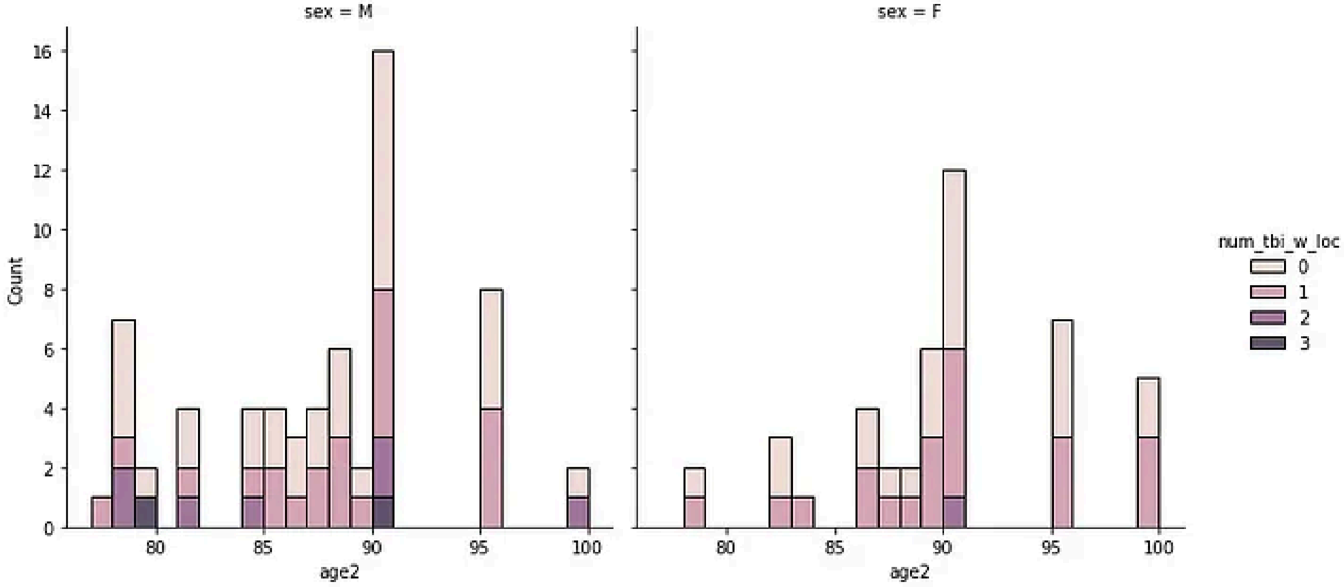


Figure 9. Number of donors who are male (left) and female (right) by count of traumatic brain injuries (TBI)

Producing a model based on unbalanced data can create unintended consequences. For instance, predicting someone as likely to have dementia could create a perception of health risks that leads to unjustified higher insurance premiums. Building a model using a larger sample of donors is critical for commercial use by insurance companies in underwriting LTC insurance applicants. Insurance is a highly regulated industry, and insurance companies will thoroughly vet the model to minimize the possibility of any unintended biases.

Gathering domain expertise

We spoke to a professor of human genetics at the University of Michigan who advised that the small sample size and the fact that the dataset has all residents of Washington state helps to reduce the between-group differences. Each of the experimental and control cohorts had 55 individuals, and keeping the two groups homogenous helps prevent the introduction of confounding variables. At the same time, the homogeneity creates challenges for data scientists who wish to generalize their model beyond this sample.

This expert advised that any study with human samples has an endless cascade of factors to consider as inputs. Other machine learning models have factored in age, genetic makeup, and environmental exposure in predicting dementia (Barnes et al.).

Lastly, we received advice from this expert to rank the type of features we include in our model and explain which features we left out, so that we are transparent about the features our model does not account for.

Conclusion

Although we set out to create a model that identified genes associated with dementia or the risk of dementia based on brain tissue samples, we have learned that developing a model ready for commercial use by the insurance sector would need further research into how to prevent overfitting and how to test the model on other types of tissue data.

What our modeling effort has affirmed is that machine learning models can provide another avenue to identify genes possibly contributing to dementia. Finding significant correlations is one we hope to reach as we continue to apply our machine learning detective skills.

Statement of Work

Each team member contributed their strengths to this project. Pedro Hermon led on model selection, testing, and evaluation. He also contributed visuals created in Altair, Matplotlib, NetworkX, and other Python libraries. Jonathan Yoon led the development of our Github site and researched and recommended a blog platform. Stephanie Tsao led on project management, soliciting domain expertise, and the writing of the blog. All team members helped in cleaning code and contributing to the blog.

Works Cited

Allen Institute for Brain Science, University of Washington Medicine, and Kaiser Permanente Washington Health Research Institute. “Aging, Dementia, and TBI Study [dataset].” Available from aging.brain-map.org. RRID:SCR_014554 | Primary publication: Miller J. A., et al. “Neuropathological and transcriptomic characteristics of the aged brain.” eLife, 2017;6:e31126. https://doi.org/10.7554/eLife.31126.

Barnes, DE, et al. “Development and Validation of eRADAR: A Tool Using EHR Data to Detect Unrecognized Dementia.” *Journal of the American Geriatrics Society*, vol. 68, no. 1, 2020, pp. 103–111. doi: 10.1111/jgs.16182. PMID: 31612463; PMCID: PMC7094818.

Blanchard, J.W., et al. “APOE4 impairs myelination via cholesterol dysregulation in oligodendrocytes.” Nature, vol. 611, 2022, pp. 769–779. doi: 10.1038/s41586–022–05439-w.

Javeed, A., et al. “Machine Learning for Dementia Prediction: A Systematic Review and Future Research Directions.” *Journal of Medical Systems*, vol. 47, no. 17, 2023, https://doi.org/10.1007/s10916-023-01906-7.

Manly, J. J., et al. “Estimating the Prevalence of Dementia and Mild Cognitive Impairment in the US: The 2016 Health and Retirement Study Harmonized Cognitive Assessment Protocol Project.” JAMA Neurol, vol. 79, no. 12, 2022, pp. 1242–1249. doi: 10.1001/jamaneurol.2022.3543. PMID: 36279130; PMCID: PMC9593315.

Mendez, MF. "What is the Relationship of Traumatic Brain Injury to Dementia?" *Journal of Alzheimer's Disease,* vol. 57, no. 3, 2017, pp. 667–681. doi: 10.3233/JAD-161002. PMID: 28269777.

Müller, Andreas C. and Guido, Sarah. Introduction to Machine Learning with Python. O'Reilly Media, 2017.

Nahm FS. Receiver operating characteristic curve: overview and practical use for clinicians. Korean J Anesthesiol. 2022 Feb;75(1):25–36. doi: 10.4097/kja.21209. Epub 2022 Jan 18. PMID: 35124947; PMCID: PMC8831439.

Rappaport, N., Twik, M., Plaschkes, I., Nudel, R., Iny Stein, T., Levitt, J., Gershoni, M., Morrey, C.P., Safran, M., and Lancet, D. "MalaCards: The Human Disease Database." *Nucleic Acids Research*, 2016, DOI: 10.1093/nar/gkw1012, https://www.malacards.org/.

U.S. Centers for Medicare and Medicaid Services. "Long-Term Care Coverage." Medicare.gov, https://www.medicare.gov/coverage/long-term-care, Data of Access: November 30, 2023.

Zazo, Javier and Protopapas, Pavlos. "SVMs, logistic regression and deep learning." Harvard University, Feb. 28, 2018, https://scholar.harvard.edu/files/javierzazo/files/svms.pdf.

Zhang B., et al. "Integrated systems approach identifies genetic nodes and networks in late-onset Alzheimer's disease." *Cell,* vol. 153, no. 3, 2013, pp. 707–720. PMID: 23622250.

Data Science Genetics Dementia Machine Learning Ai

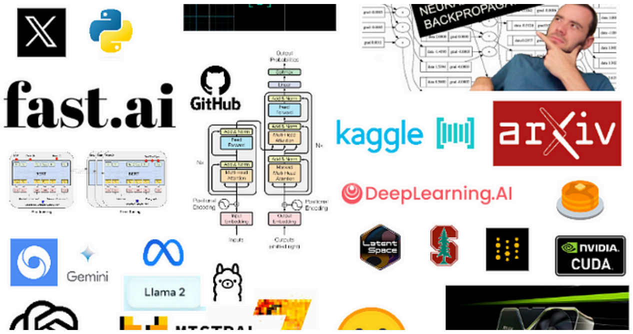
Machine Learning Python

Written by Stephanie Tsao

Edit profile

1 Follower

Recommended from Medium



Benedict Neo in bitgrit Data Science Publication

Roadmap to Learn AI in 2024

Mariyatta Louis

Machine Learning Introduction

A free curriculum for hackers and programmers to learn AI

11 min read · Mar 11, 2024

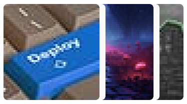
👏 9.6K 💬 104

Machine learning is a subfield of artificial intelligence that provides machines the...

✦ · 4 min read · Feb 13, 2024

👏 164 💬 3

Lists



Predictive Modeling w/ Python

20 stories · 1071 saves

Practical Guides to Machine Learning

10 stories · 1286 saves

Coding & Development

11 stories · 549 saves

ChatGPT prompts

47 stories · 1383 saves

Cristian Leo in Towards Data Science

The Math Behind Neural Networks

Dive into Neural Networks, the backbone of modern AI, understand its mathematics,...

28 min read · Mar 29, 2024

👏 2.1K 💬 15

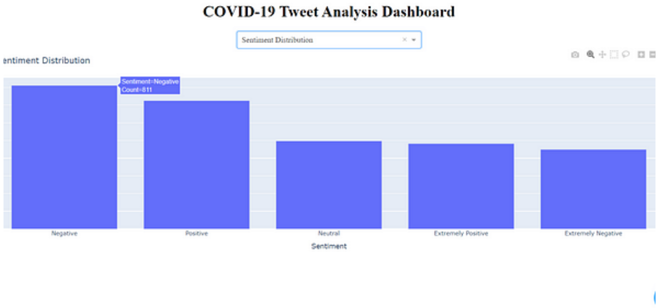
Osamaabdelaziem

Unsupervised Anomaly Detection for Predictive Maintenance

Introduction

5 min read · Mar 20, 2024

👏 11 💬



Mamoon Khan

Title: Unveiling Insights: Exploring Corona and Homicide Data...

Introduction

5 min read · 4 days ago

Marco Del Coco

TimesFM: the missing Foundation Models for Time Series...

Foundation Models for Time Series forecasting are still missing, or at least it was...

✦ · 8 min read · Mar 18, 2024

63

See more recommendations