|  |
| --- |
| CS5785: Final Project |
| Final Report |

|  |
| --- |
| Saloni Vishwakarma and Jessica White  12-14-2020 |

Contents

[Abstract: 3](#_Toc58860007)

[Introduction: 3](#_Toc58860008)

[Background: 3](#_Toc58860009)

[Methods: 3](#_Toc58860010)

[Experimental analysis: 5](#_Toc58860011)

[Ordinary least squares regression and Lasso for inference 5](#_Toc58860012)

[Support vector regression 6](#_Toc58860013)

[Gradient boosting regression 6](#_Toc58860014)

[Discussion and prior work: 6](#_Toc58860015)

[Conclusion: 6](#_Toc58860016)

[Supplementary Figures 7](#_Toc58860017)

[Figure 1: % viability for various summary statistics and tumors vs. sentiment analysis. 7](#_Toc58860018)

[Figure 2: Percent viability by cell line and tumor type. 7](#_Toc58860019)

[Figure 3: Percent viability of MEK inhibitors in PN cell lines 8](#_Toc58860020)

[Figure 4: Histogram of percent viability of drug/condition combinations in abstracts. 8](#_Toc58860021)

[Figure 5a: Initial Word Frequency and Wordcloud during NLP Pre-Processing Steps 9](#_Toc58860022)

[Figure 5b: Updated Word Frequency and Wordcloud during NLP Pre-Processing Steps 9](#_Toc58860023)

[Figure 6a: PCA for various values of M using binary bag-of-words representation 10](#_Toc58860024)

[Figure 6b: PCA for various values of M using count bag-of-words representation 11](#_Toc58860025)

[Figure 6c: PCA for various values of M using TF-IDF representation 12](#_Toc58860026)

[Figure 7a: Scaled cell viability compared to actual (SVR) 13](#_Toc58860027)

[Figure 7b: Results of GridSearch optimized SVR using binary BoW matrix and scaled and unscaled data 13](#_Toc58860028)

[Figure 8a: GridSearch Results for Support Vector Regressor, MSE and SD (Binary BoW, scaled outcome, RBF kernel, C = 100, Gamma: Auto) 14](#_Toc58860029)

[Figure 8b: GridSearch Results for Support Vector Regressor, MSE and SD (Count BoW, scaled outcome, RBF kernel, C = 100, Gamma: Auto) 14](#_Toc58860030)

[Figure 8c: GridSearch Results for Support Vector Regressor, MSE and SD (TF-IDF, scaled outcome, sigmoid kernel, C = 10,000, Gamma: 0.001) 15](#_Toc58860031)

[Figure 9a: GridSearch Results for Gradient Boosting Regressor, MSE and SD (Binary BoW, scaled outcome, LR = 0.05, Max Depth = 2, Estimators = 1,000, Subsample = 0.5) 16](#_Toc58860032)

[Figure 9b: GridSearch Results for Gradient Boosting Regressor, MSE and SD (Count BoW, scaled outcome, LR = 0.05, Max Depth = 3, Estimators = 500, Subsample = 0.5) 18](#_Toc58860033)

[Figure 9c: GridSearch Results for Gradient Boosting Regressor, MSE and SD (TF-IDF, scaled outcome, LR = 0.1, Max Depth = 3, Estimators = 500, Subsample = 0.5) 20](#_Toc58860034)

[Figure 10a: Comparison of cross-validation error (linear regression, SVR, and GBR) 22](#_Toc58860035)

[Figure 10b: Comparison of cross-validation error (Lasso, SVR, and GBR) 22](#_Toc58860036)

[Figure 10a: Predicted vs. actual % viability in pNF cell lines by agent and mechanism (Linear Regression) 24](#_Toc58860037)

[Figure 10b: Predicted vs. actual % viability in pNF cell lines by agent and mechanism (Support Vector Regression) 24](#_Toc58860038)

[Figure 10c: Predicted vs. actual % viability in pNF cell lines by agent and mechanism (Gradient Boosting Regression) 25](#_Toc58860039)

[Figure 11: Delta between actual and predicted in pNF cell lines by drug and mechanism 26](#_Toc58860040)

[References 28](#_Toc58860041)

# Abstract:

[TO COME]

# Introduction:

The ease of predicting which drugs are likely to be effective in various oncology settings depends largely on the quality of cell lines and the availability of open-source screening databases that test the ability of numerous chemical compounds to halt growth (cytostatic) or even kill tumor cells (cytotoxic). For many conditions, cell lines are not particularly predictive of response *in vivo.* Even fewer conditions have both cell lines that can reliably predict drug responses in humans and databases of drug responses that can be mined. However, many compounds have been tested in both preclinical and clinical experiments in a non-systematic manner, with the data published in academic articles. Indeed, the more promising a compound, the more likely it is to appear in a variety of publications. As such, we attempted to design an algorithm that was able to predict performance of drugs in certain cancer cell lines by performing natural language processing (NLP) on the corpus of journal articles that mention the drug and cancer indication of interest. We would conceive of such a tool being used by academic investigators or industry sponsors to rapidly identify the most promising compounds tested in a given oncology indication even in the absence of databases of cell line-specific potency.

# Background:

We began with a database of cell line responses, as measured by percent viability, supplied by the Children’s Tumor Foundation (CTF) to assist in the discovery of drugs to treat Neurofibromatosis Type 1 (NF1), Neurofibromatosis Type 2 (NF2), and Schwannomatosis.(*1, 2*) These are all autosomal dominant genetic conditions that are tumor predisposition syndromes. Some of the most common tumor types that develop in these patients are schwannomas and meningiomas in NF2 and plexiform neurofibromas (PN or pNF) in NF1, which are tumors of the lining of the central or peripheral nervous system. The CTF database contains drug response data for all 3 of these tumor types. These tumors are mostly benign, meaning they usually do not possess the ability to metastasize. However, they can be highly painful, disfiguring, and debilitating. Currently, there is only a single FDA-approved drug for the treatment of PNs, selumetinib, though other drugs in the same class, MEK inhibitors, have shown promise preclinically and clinically, as well.(*3*) No drugs are currently available to effectively treat meningiomas, and the drug most commonly used to treat schwannomas, bevacizumab, is essentially cytostatic rather than cytotoxic and cannot be used for prolonged periods of time.(*4, 5*) As such, we found it important to identify additional compounds in the literature that might be promising for the treatment of meningiomas, schwannomas, and PNs. We also had access to a list of articles and corresponding abstracts curated via Elsevier’s Scopus database. A total of 5,702 articles published between 2010 and 2019 were selected on the basis that the terms “neurofibromatosis” or “schwannomatosis” occurred in their title, abstract, or keywords. Overall, our main aim was to take real-world data and apply NLP pre-processing and algorithms to assess the promise of various compounds in these indications.

# Methods:

First, we assessed how the abstracts could be labeled to indicate a promising drug-tumor combination. Our intention was to then build a supervised model to using the abstracts and cell viability data as features to identify drugs that did indeed seem to lead to tumor regression. Initially we investigated 2 sentiment detection algorithms, VADER and TextBlob to see if either or a combination of the 2 could be used to label the abstracts as positive or negative.(*6*) These algorithms use pre-specified dictionaries to identify the sentiment (i.e., the relative positive or negative implications) of a given text on a continuous scale. To see if we could establish concordance between the results per sentiment analysis and the cell viability results per the CTF database, we first used the DrugExplorer database to map the internal CTF identifier to commonly used drug names, predominantly International Nonproprietary Names. Next, we selected abstracts on the basis that they mentioned either the terms “meningioma,” “schwannoma,” or “plexiform” at least once to classify the abstract as relating to that specific tumor type. Finally, we took the drug names from the cell line data in each of those 3 tumor types and searched the abstracts labeled with the relevant tumor types again for a single occurrence of the drug so that we could establish a corpus of drug and tumor type combinations. This resulted in 144 abstracts that mentioned drugs and tumor combinations that had been tested in the CTF drug response database on which we then used VADER and TextBlob to perform sentiment analysis.

To establish the consensus percent viability of the identified drugs in the relevant tumor cell lines, we took summary statistics from all cell lines tested for each tumor type, including the median, mean, minimum, and maximum. Given that the VADER and TextBlob scores fall within different ranges, we standardized the scores such that they had a mean of zero and a standard deviation of 1. As shown in [Fig. 1](#_Figure_1:_%), there is no apparent correlation between either of these sentiment analysis approaches and the cell viability data. Indeed, upon manual inspection, we discovered a number of abstracts that were clearly semantically positive labeled as negative. This is attributable to the sentiment analysis dictionary categorizing words that were positive in the context of effect on tumor cell lines as negative, such as regression or reduction. This suggested that building an algorithm that could identify words likely to indicate positive outcomes in the context of oncology research would indeed be helpful as more general sentiment analysis tools did not yield fruitful results. Additionally, another labeling approach would be needed to identify promising drugs.

We then considered using the percent viability by drug and tumor type to label the abstracts. Generally, lower percent viability indicates more promising results, with values significantly below 100% indicating that a compound has not merely cytostatic but cytotoxic potential. In [Fig. 2](#_Figure_2:_Percent), which shows boxplots of the percent viability from each tumor cell line colored by tumor type from which the cell line is derived, 3 of the 4 meningioma cell lines and 3 of the 3 schwannoma cell lines show a number of compounds with strong cytotoxic potential. However, none of the PN cell lines seemed to show robust cytotoxicity with the interquartile ranges centered tightly around 100%. Indeed, as shown in [Fig. 3](#_Figure_3:_Percent), the percent viability in PN cell lines of MEK inhibitors, which have demonstrated potent cytotoxic effects clinically in PNs as a class, is not indicative that these compounds are cytotoxic or even cytostatic. This suggests that PN cell lines are not particularly predictive of likely activity of a drug. As such, we decided that we would use the abstracts related to PNs as testing data and the abstracts related to meningiomas and schwannomas as the training data. In [Fig. 4](#_Figure_4:_Histogram), we can see the distribution of labels by condition of the 144 abstracts we collected is largely consistent with the overall distribution.

We had now clearly defined the problem, namely to use the abstracts related to meningioma and schwannoma to predict the percent cell viability by drug and apply this model to predict the most promising drugs in the setting of PN. We then performed natural language processing (NLP) pre-processing steps, including removal of frequently occurring stop words that lack informative meaning, tokenization of the abstracts into single words, and lemmatization or grouping together the inflected forms of a word. To normalize the texts, we expanded any contractions, converted accented characters to ASCII characters, and converted all texts to lowercase. Additionally, we removed URLs, HTML tags, punctuation, and extra whitespaces. We also noticed that the sentiment could vary over the course of the abstract. For example, abstracts frequently begin by characterizing the unmet medical need in the condition of interest, which is often relatively negative. Only once the experiment is characterized and the drug of interest is mentioned is relevant information regarding the effectiveness of the drug mentioned. As such, we decided to extract only ten words flanking each occurrence of the drug name to create a modified version of the abstract.

Concurrently, we removed stop words. Initially, we exclusively used the stop words corpus available via the nltk module to remove stop words, but we later observed that words such as “neurofibromatosis,” “nf,” “meningiomas,” “schwannomas,” and other content-specific terms appeared very frequently further downstream in our analysis as shown in [Fig. 5a](#_Figure_5a:_Initial_1). Since we are working with a specific disease and conditions that we have already initially filtered our abstracts, drugs, and cell lines’ data for, we augmented the nltk stop word dictionary with such technical terms along with drug names. This way, we collect words, which can be viewed in [Fig. 5b](#_Figure_5b:_Updated), that will potentially hold *more* value in our machine learning algorithms. Next, we tokenized the normalized text by splitting up the larger bodies of text into individual units called tokens, or words. Finally, we the remaining text was lemmatized, which removes inflectional endings only and returns the base or dictionary form of a word.

Across the processed abstracts, there were 86 unique drug/condition pairs, indicating multiple abstracts relevant to some drug and condition combinations. While we could have assessed each abstract individually and then used an ensemble technique (e.g., averaging or voting) to aggregate the predicted percent viability for a given drug in a specific condition following prediction, we decided to aggregate all abstracts relating to these 86 pairs into a single pre-processed document. For each drug/condition pair, we have the following information in the form of a dataframe: the union of the words extracted from abstracts relevant to that pair, the number of those abstracts involved, and percent viability statistics (median, mean, min, max). This dataframe is appropriately structured to address our main question with text information from our abstracts along with our target information, percent cell viability, arranged by drug/condition.

# Experimental analysis:

## Numeric representation of text

The first way we formally assessed the data after pre-processing was by converting our text to numerical feature vectors using bag-of-words (BoW), both with binary and count matrices. and term frequency-inverse document frequency (TF-IDF) methods. BoW provides a count of the words occurring in the training set, either in a binary manner or in a count reflecting the total number of occurrences in the document. TF-IDF assigns each word in the document a weight that is directly proportional to the relative frequency of the word in a document and inversely proportional to the number of documents in the corpus that contain that word. This is another mechanism meant to diminish the importance of stop words given that words that occur very frequently across the corpus will have a reduced weight by TF-IDF. To compare these methods and select the relevant hyperparameter, M, which represents the minimum frequency percentage cut-off for the BoW representations, we used Principal Component Analysis (PCA) to visually assess whether these methods were able to effectively separate the drug condition combinations into clusters of similar percent viabilities. In the final feature matrices, we included these representations along with the feature “freq” representing the number of abstracts that were included in the revised document.

We represented the first and second principal components in a 2D plot and the first, second, and third principal components in a 3D plot in [Fig. 6a](#_Figure_6a:_PCA), [6b](#_Figure_6b:_PCA), and [6c](#_Figure_6c:_PCA). We also measured the mean and standard deviation of the Euclidean distance between points using the first 3 principal components as a proxy for separation between points. The binary BoW representation ([Fig 6a](#_Figure_6a:_PCA)) was highly sensitive to the choice of M, and the furthest separation between points was seen for M = 0.05. The count BoW representation ([Fig 6b](#_Figure_6b:_PCA)) remained fairly stable throughout the range of M tested, but the separation between points was seen at M = 0.01. For the TF-IDF representation, since we have a fixed vocabulary, we selected M = 0.05, consistent with the M used in the binary BoW. While all methods conferred some degree of separation for a few drug/condition pairs, the 3D representations still showed the points quite tightly clustered. It is possible that 3 principal components are still not sufficient to capture the variation in the underlying data, and a better approach might be determining an elbow point via a Scree plot and using that many principal components in the Euclidean distance measurements instead. We used all 3 representations in the subsequently discussed algorithms.

## Scaling data

We then wanted to determine whether we should scale the outcome data. Given that the word matrices represent either a binary or count variable or are already weighted, in the case of TF-IDF, we chose not to further scale the feature data. We anticipated a Support Vector Regression (SVR) was likely to be perform well, so we used this algorithm to assess the utility of scaling the data. Initially, we scaled the training (meningioma, schwannoma) and test (pNF) outcomes separately. However, as we saw in [Fig 4](#_Figure_4:_Histogram), the ranges of these 2 data sets were fairly disparate, leading to unpredictable and biologically unlikely outcomes, such as strongly negative cell viability estimates. Since our ultimate goal was to generate “adjusted” cell viability estimates for the pNF cell lines that more accurately reflect the efficacy of certain compounds based on the literature, we pooled the training and test data to scale the outcome measure. The resulting scaled outcome measures shown in [Fig 7a](#_Figure_7a:_Scaled). We then ran a GridSearch-optimized SVR on either the scaled or unscaled data and compared the resulting predicted percent cell viabilities for the training and test data ([Fig 7b](#_Figure_7b:_Results)). Overall, the outcomes were quite similar regardless of scaling and generally recapitulated the shape of the data with the pNF/test data shifted downwards, as was our goal. Our analyses hereafter use scaled data as our outcomes but similar results were generally seen with unscaled data.

## Ordinary least squares (OLS) and Lasso regression

We first ran OLS and Lasso regressions using sci-KitLearn’s LinearRegression and Lasso implementations, respectively. For Lasso, we implemented a GridSearch where the alpha parameter fell between 0 (equivalent to an OLS regression) and 1 and tolerance (tol) between 10-5 and 10-3. For all three input matrices, the optimal parameters were alpha = 0.001 and tol = 10-5. We also used the results from Lasso to display the remaining non-zero coefficients and probe feature importance ([Fig 8](#_Figure_8:_Common)).

## Support vector regression

Then, we ran an SVR using sci-KitLearn’s implementation. We used a GridSearch to optimize the parameters; we tested γ between 10-4 and 0.5 as well as the default “auto,” which uses 1/n\_features and “scale,” which uses 1 / (n\_features \* X.var()), linear, polynomial, sigmoid, and RBF kernels, and C between 1 and 10,000. For the BoW binary and BoW count representations, the optimal parameters were γ = auto, C = 100, and an RBF kernel. For the TF-IDF representation, the optimal parameters were γ = 0.001, C = 10,000, and a sigmoid kernel. An SVM with a sigmoid kernel is equivalent to a 2-layer perceptron. Plots from the SVR GridSearch are shown in [Fig. 9a](#_Figure_8a:_GridSearch), [9b](#_Figure_8b:_GridSearch), and [9c](#_Figure_8c:_GridSearch).

## Gradient boosting regression (GBR)

Finally, we ran a GBR using sci-KitLearn’s tree-based GradientBoostingRegressor. In the GridSearch, we tested learning rates ranging from 0.05 to 1, number of estimators between 10 and 1,000, subsampling between 25% and 100% of the data, a max depth of between 2 and 10, and max features either equal to n\_features (auto) or equal to the square root of the number of features (sqrt). For all representations, the optimal max feature parameter was “auto” and the optimal subsample was 50%. The optimal learning rate was 0.05 for binary and count BoW and 0.1 for TF-IDF. The optimal max depth was 2 for binary BoW and 3 for count BoW and TF-IDF. The optimal number of estimators was 1,000 for the binary BoW and 500 for the count BoW and TF-IDF. Plots from the GBR GridSearch are shown in [Fig. 10a](#_Figure_9a:_GridSearch), [10b](#_Figure_9b:_GridSearch), and [10c](#_Figure_9c:_GridSearch).

## Comparison of model performance

Since our test data was labeled with percent viability estimates that did not accurately reflect the clinical promise of these drugs, the utility of comparisons of performance on the test data was limited. Indeed, the mean-squared error (MSE) in Table 1 appears quite high compared to the training MSE, seeming to indicate the model is overfitting the training data and not generalizing adequately. Therefore, we pursued a two alternative approaches to assess model performance: 1) evaluating the estimated prediction error, which includes some element of generalization error, by comparing the MSE on the k-fold cross-validation sets, 2) gauging performance on the test data more qualitatively by seeing which algorithms best recovered the mechanisms and compounds that have shown promise clinically.

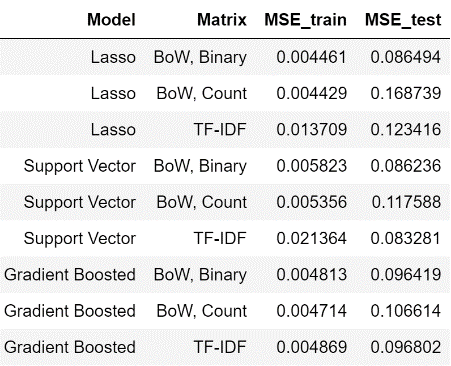


Table 1: Training and Test MSE for Various Models

[Fig. 11a](#_Figure_10a:_Comparison) shows the strip charts and boxplots of performance using the different representation methods and OLS, SVR, or GBR algorithms. The OLS using a BoW count representation was excluded as this model resulted in extremely high MSE, skewing the plot. Interestingly, the other OLS methods appear fairly robust when compared to the SVR and GBR-based models. Generally, the GBR-based models produced the lowest MSE, followed by SVR. The BoW binary representations produced models with the lowest MSE in each algorithm class. [Fig 11b](#_Figure_10b:_Comparison) shows the same figure but with Lasso regressions instead of an OLS regressions. The performance of the Lasso and the SVR models are quite similar, but GBR retains the highest apparent performance and lowest generalization error.

To qualitatively compare the models’ performances, we plotted predicted vs. actual percent viability in pNF cell lines in [Fig. 12a](#_Figure_11a:_Predicted) (Lasso), [12b](#_Figure_11b:_Predicted) (SVR), and [12c](#_Figure_11c:_Predicted) (GBR). The dashed line represents the line where the predicted percent viability is equivalent to the actual. We are interested in drugs that deviate from the line the furthest, indicating our model predicts a much lower percent viability (favorable outcome) based on the abstract texts as compared to the experimentally determined percent viability. As previously, mentioned, MEK inhibitors are by far the most promising class of drugs, so we would expect those to emerge as top hits. Similarly, imatinib has shown activity clinically in pNF, and a study of its next-generation successor, nilotinib, is ongoing.(*7*) Notably, while vincristine appeared to be a top hit in a number of models by this metric, particularly when a binary BoW matrix was used to represent the texts, this is a false positive. It results because it is part of the standard-of-care chemotherapy regimen in optic pathway gliomas (OPG), yet another tumor that results from a germline NF1 mutation, and its occurrence is in an abstract mentioning both pNF and OPG investigational drugs.(*8*)

Ultimately, it appears that the SVR and the Lasso regression using the count BoW representation best reflects the promise of MEKi as a class ([Fig 12b](#_Figure_11b:_Predicted) and [Fig 12a](#_Figure_11a:_Predicted)). These models also recognize the relative importance of imatinib and nilotinib. Interestingly, while the GMB methods appeared to perform the best when comparing the cross-validation MSE, they failed to recognize that MEKi were the most promising class, per the abstracts ([Fig 12c](#_Figure_11c:_Predicted)). When comparing results across algorithm and representation ([Fig. 13](#_Figure_12:_Delta)), five drugs stand out as the most promising: imatinib, selumetinib, trametinib, PD-0325901, and nilotinib. Notably, all of those agents have been deemed sufficiently promising to be tested in clinical trials.

# Discussion and prior work:

Employing machine learning algorithms that process texts (i.e. drug reviews, scientific publications) to provide an interpretation of drug performance has been deeply explored in the past few years. A common approach found in recently published work was to investigate textual patient reviews of drugs via sentiment analysis to predict patient satisfaction. Using the Drug Review Dataset from the UCI Machine Learning repository, Vijayaraghavan and Basu (2020) implemented a supervised classification algorithm to predict the class of overall drug rating which was represented by a range of 1 to 10.(*9*) Using different text embeddings such as TFIDF and count vectors, and performing grid search for hyperparameter tuning, they concluded that after neural networks and SVMs perform drastically better than logistic regression models. In this particular study, the researchers divided the data set based on condition and reported that their best SVM performed with 84% accuracy for the condition of depression. Similar analyses were completed by many others and reported on public blogging websites, such as *Medium* and *Towards Data Science*. Another interesting approach taken by Na and Kyaing (2015) was to investigate various text representations for SVMs, including bag of words, independent vs dependent clause separation, and word negation.(*10*) Their goal was to build a multi-class classifier to predict sentiment using textual public opinions of drugs from WebMD, and ultimately, they reported an accuracy of 66% for their best performing SVM.

We build upon previously published work by using and strategically combining unprocessed datasets supplied by the Children’s Tumor Foundation instead of using preprocessed or a toy dataset. Instead of classification, we undertook designing an algorithm to perform regression and predict performance of drugs using a corpus of journal articles that mention the drug and symptom indication of interest. Similar to the methods from the literature research mentioned above, we performed robust grid search to tune our hyperparameters, compared multiple machine learning algorithms such as support vector regression, lasso regression, and gradient boosting regression. Finally, we visualized cross-validated data across all methodologies to provide insight into the anticipated generalization error for our testing set.

# Conclusion:

# Supplementary Figures

## Figure 1: % viability for various summary statistics and tumors vs. sentiment analysis.

Calendar

Description automatically generated

## Figure 2: Percent viability by cell line and tumor type.

Chart, box and whisker chart

Description automatically generated

## Figure 3: Percent viability of MEK inhibitors in PN cell lines

Chart, box and whisker chart

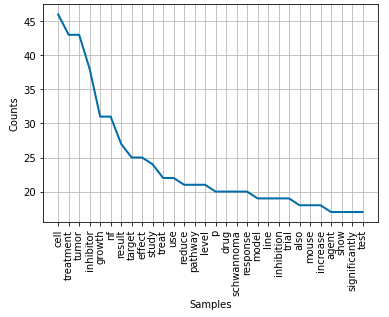
Description automatically generated

## Figure 4: Histogram of percent viability of drug/condition combinations in abstracts.

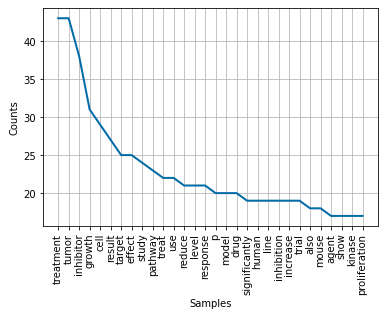
Chart, histogram

Description automatically generated

## Figure 5a: Initial Word Frequency and Wordcloud during NLP Pre-Processing Steps



## Figure 5b: Updated Word Frequency and Wordcloud during NLP Pre-Processing Steps



## Figure 6a: PCA for various values of M using binary bag-of-words representation

Chart, scatter chart

Description automatically generatedDiagram

Description automatically generated

## Figure 6b: PCA for various values of M using count bag-of-words representation

Chart, scatter chart

Description automatically generatedDiagram

Description automatically generated

## Figure 6c: PCA for various values of M using TF-IDF representation

Chart, scatter chart

Description automatically generated

A picture containing text

Description automatically generated

## Figure 7a: Scaled cell viability compared to actual (SVR)

Chart, histogram

Description automatically generated

## Figure 7b: Results of GridSearch optimized SVR using binary BoW matrix and scaled and unscaled data

Chart, histogram

Description automatically generated

## Figure 8: Common coefficients using various representations and GridSearch-optimized Lasso Regression

Diagram, venn diagram

Description automatically generated

## Figure 9a: GridSearch Results for Support Vector Regressor, MSE and SD (Binary BoW, scaled outcome, RBF kernel, C = 100, Gamma: Auto)

Chart, line chart

Description automatically generated

Chart, line chart

Description automatically generated

## Figure 9b: GridSearch Results for Support Vector Regressor, MSE and SD (Count BoW, scaled outcome, RBF kernel, C = 100, Gamma: Auto)

Chart

Description automatically generated

Chart, line chart

Description automatically generated

## Figure 9c: GridSearch Results for Support Vector Regressor, MSE and SD (TF-IDF, scaled outcome, sigmoid kernel, C = 10,000, Gamma: 0.001)

Chart, line chart

Description automatically generated

Chart, line chart

Description automatically generated

## Figure 10a: GridSearch Results for Gradient Boosting Regressor, MSE and SD (Binary BoW, scaled outcome, LR = 0.05, Max Depth = 2, Estimators = 1,000, Subsample = 0.5)

Chart, line chart, polygon

Description automatically generated

Chart, line chart, polygon

Description automatically generated

## Figure 10b: GridSearch Results for Gradient Boosting Regressor, MSE and SD (Count BoW, scaled outcome, LR = 0.05, Max Depth = 3, Estimators = 500, Subsample = 0.5)

Chart, line chart, polygon

Description automatically generated

Chart, line chart, polygon

Description automatically generated

## Figure 10c: GridSearch Results for Gradient Boosting Regressor, MSE and SD (TF-IDF, scaled outcome, LR = 0.1, Max Depth = 3, Estimators = 500, Subsample = 0.5)

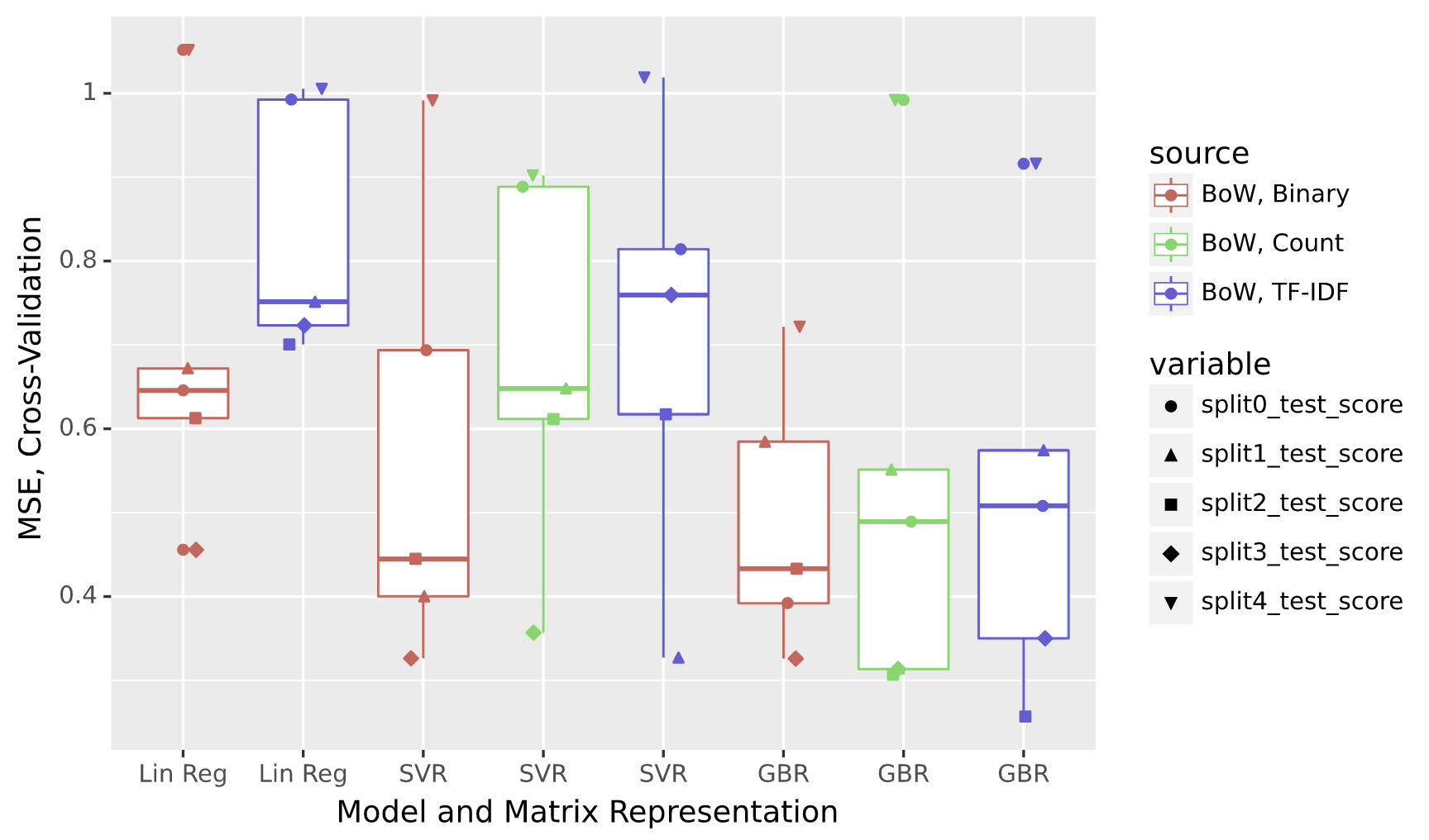
Chart, line chart, polygon

Description automatically generated

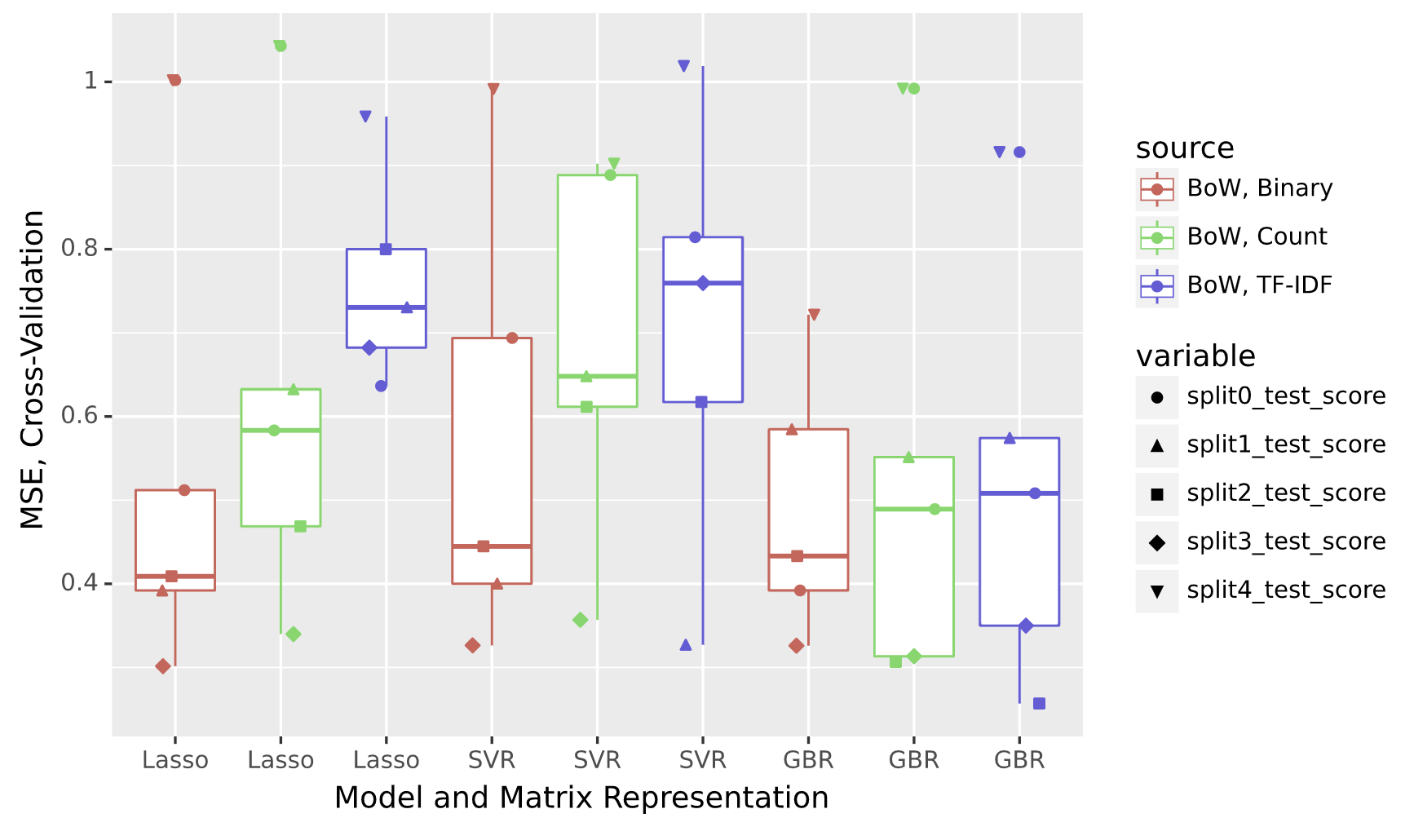
Chart, line chart, polygon

Description automatically generated

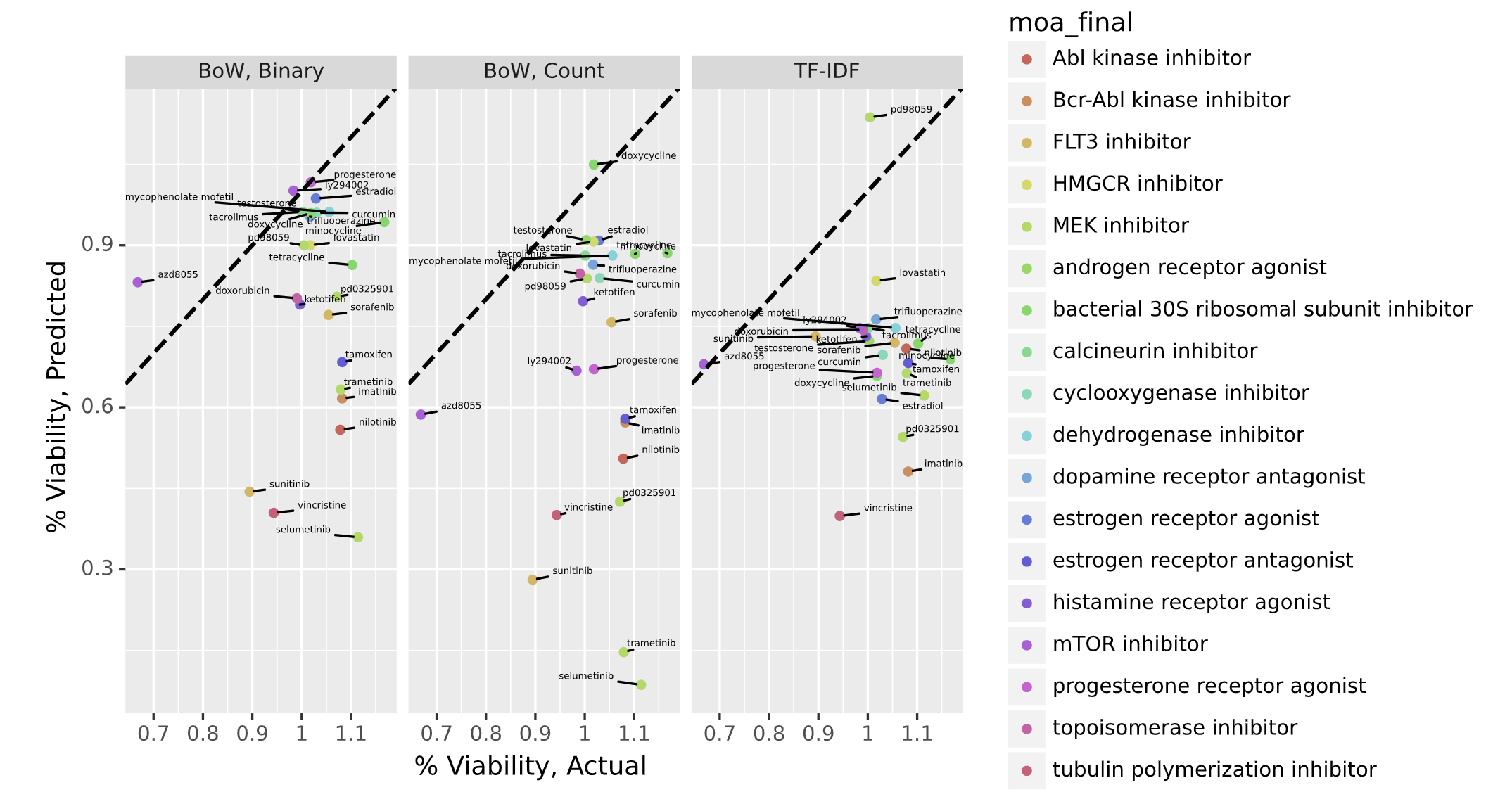
## Figure 11a: Comparison of cross-validation error (linear regression, SVR, and GBR)



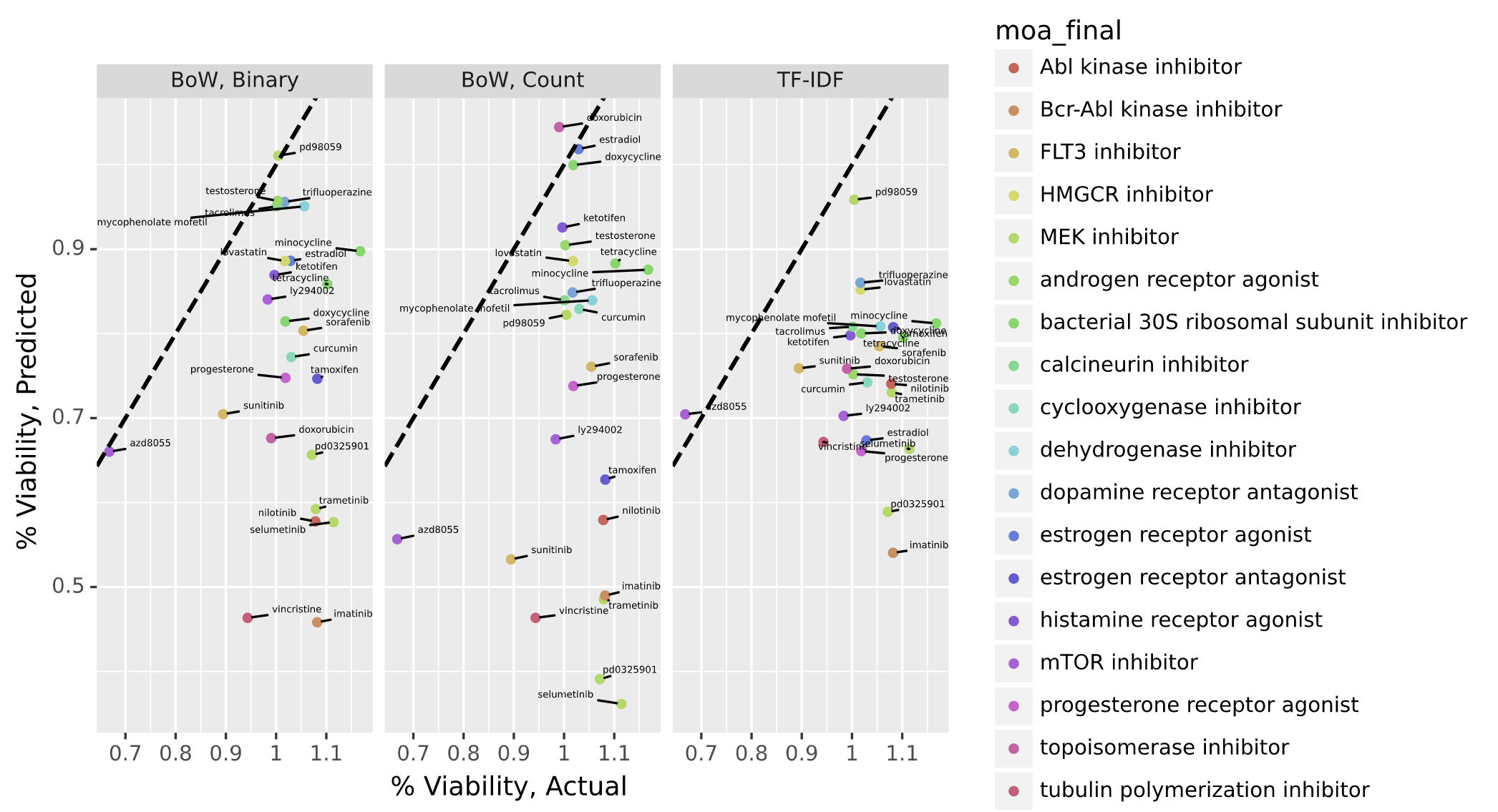
## Figure 11b: Comparison of cross-validation error (Lasso, SVR, and GBR)



## Figure 12a: Predicted vs. actual % viability in pNF cell lines by agent and mechanism (Lasso Regression)



## Figure 12b: Predicted vs. actual % viability in pNF cell lines by agent and mechanism (Support Vector Regression)



## Figure 12c: Predicted vs. actual % viability in pNF cell lines by agent and mechanism (Gradient Boosting Regression)

Chart

Description automatically generated

## Figure 13: Delta between actual and predicted in pNF cell lines by drug and mechanism

Chart

Description automatically generated

# References

1. M. Ferrer *et al.*, Pharmacological and genomic profiling of neurofibromatosis type 1 plexiform neurofibroma-derived schwann cells. *Sci Data* **5**, 180106 (2018).

2. R. Allaway *et al.*, Traditional and systems biology based drug discovery for the rare tumor syndrome neurofibromatosis type 2. *PLoS One* **13**, e0197350 (2018).

3. Selumetinib in Children with Inoperable Plexiform Neurofibromas. *N Engl J Med* **383**, 1290 (2020).

4. F. Nigim, H. Wakimoto, E. M. Kasper, L. Ackermans, Y. Temel, Emerging Medical Treatments for Meningioma in the Molecular Era. *Biomedicines* **6**, (2018).

5. S. Rosahl, C. Bohr, M. Lell, K. Hamm, H. Iro, [Diagnosis and Management of Vestibular Schwannomas - An Interdisciplinary Challenge]. *Laryngorhinootologie* **96**, S152-S182 (2017).

6. C. Gilbert, H. Eric, paper presented at the *Eighth International Conference on Weblogs and Social Media (ICWSM-14). Available at (20/04/16)* [*http://comp.social.gatech.edu/papers/icwsm14.vader.hutto.pdf*](http://comp.social.gatech.edu/papers/icwsm14.vader.hutto.pdf), 2014.

7. K. A. Robertson *et al.*, Imatinib mesylate for plexiform neurofibromas in patients with neurofibromatosis type 1: a phase 2 trial. *Lancet Oncol* **13**, 1218-1224 (2012).

8. M. A. Karajannis, R. E. Ferner, Neurofibromatosis-related tumors: emerging biology and therapies. *Curr Opin Pediatr* **27**, 26-33 (2015).

9. S. Vijayaraghavan, D. Basu Sentiment Analysis in Drug Reviews using Supervised Machine Learning Algorithms. *ArXivLabs*, (2020).

10. J. Na, W. Kyaing Sentiment Analysis of User-Generated Content on Drug Review Websites. *Journal of Information Science Theory and Practice*, (2015).