Predicting Breast Cancer Diagnostics with Fine Needle Aspirate Data

Lucille Peterson

2022-11-23

From the University of California Irvine Data Set Repository, added in late 2009, is a Wisconsin diagnostic breast cancer data set. Computed from digitized images of an FNA - or Fine Needle Aspirate - of breast mass tissue, we have dozens of aggregated summary statistics of these breast mass' measurements.

There are two other features - unique patient IDs (not particularly relevant to us), and a final classification that denotes benign "B" or malignant "M" tissue. We would like to model our 30 features to determine, as best as possible, whether the tissue is benign or malignant.

Data Structure

Our data set has 569 observations with 30 features (aside from ID and classification). These are collections of multiple summary statistics for 10 measurements - mean, standard error, and "worst" or largest values - pertaining to measurements of the cell nuclei of breast mass tissue. The first 10 are mean values, the next 10 are standard error, and the "worst" values are the final 10. Of each group, these are the 10 measurements of interest;

1) Radius (mean of distances from center to perimeter, since this is not always even)
2) Texture (standard deviations of gray-scale values)
3) Perimeter
4) Area
5) Smoothness (local variation in radius lengths)
6) Compactness (perimeter^2 / area - 1.0)
7) Concavity (severity of concave portions of the contour)
8) Concave Points (number of concave portions of the contour)
9) Symmetry
10) Fractal Dimension ("coastline approximation" - 1)

While the data is comprised of a hefty sum of summary statistics rather than individual observations, thus contextualizing patterns and correlations from the data may not be as straightforward, proper investigation of the data's patterns and sound machine learning modeling approaches will yield effective means of predicting diagnostics.

Data Processing

We have no real use for the first variable and will be cutting it. Further, our features should actually be labelled appropriately, rather than being "V2" through "V32". The features will be labelled the appropriate measurement, with prefixes to denote the statistic in question; M for mean, SE for standard error, and W for "worst." Classification of "B" and "M" are translated into numeric form for correlation modeling purposes as well.

Subsets of the features for each summary statistic are also easily defined and very valuable for exploratory data analysis, especially for observing correlation. Thankfully, there is no missing data to be addressed.

Exploratory Data Analysis

First order of business, what does the correlation look like between our measurements and the diagnosis?

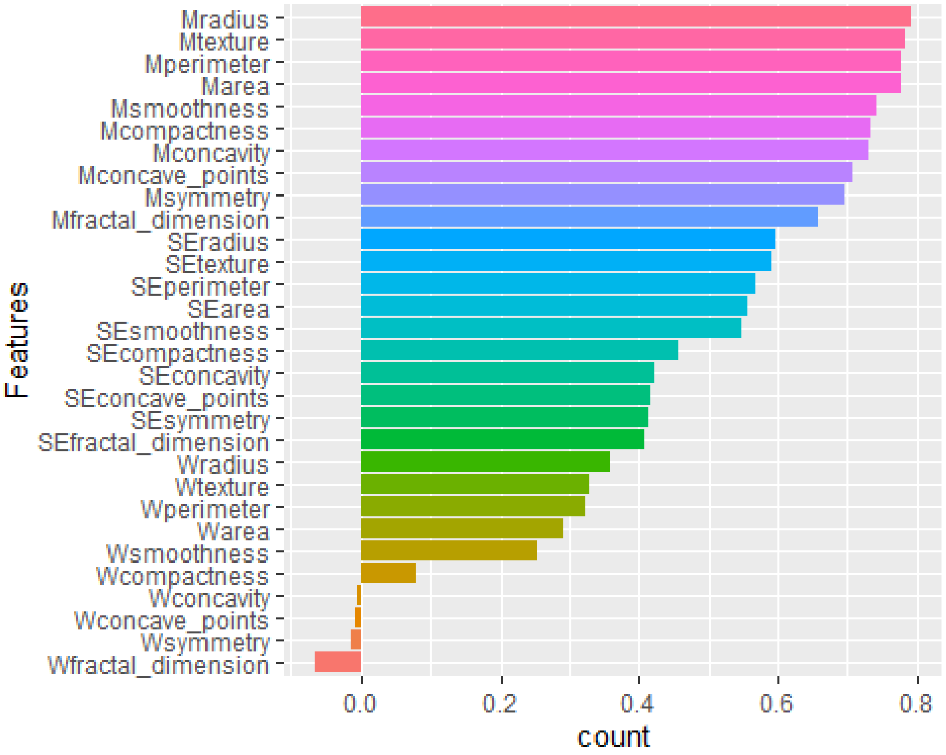

Figure 1 - Measurement correlation with diagnosis, which has been converted to a numeric form (1 for malignancy), ordered from highest numerically to lowest. Since there's only a minute amount of negative correlation here, it practically goes from high to low correlation.

Some key takeaways here, before exploring multicollinearity at all. First, negative correlation is hardly present here. Almost all across the board, increases in all of our measurements suggest an increase in the chances of a malignant, cancerous diagnosis. Considering these measurements primarily consist of breast tissue Size and Uniformity, it makes some intuitive sense that larger, more varied tissue would be more susceptible to malignancy. It is worth noting that this is related to tissue itself, as opposed to breast size.

In terms of the three different summary stats for each measurement, standard error is easily the least relevant compared to means and worst cases. The most significant standard errors appear to be for area/perimeter/radius, a series of measurements that we'd intuitively expect to be highly correlated with one another, and also appear to be some of the most important measurements in this vacuum. There's something to be said about consistency in average values, as well as worst case scenarios that would be unexpected in benign cases, both being high profile suspects for determining the likelihood of tissue being cancerous. The same case just can't be made for standard error.

We're unlikely to continue scrutinizing the standard error data of the measurements in the rest of this EDA to the same degree we'll be doing for means/worst cases, or the 10 measurements themselves. However, when we go to model, we should consider options and methodologies that can meaningfully incorporate all of it. If we were to be doing some more traditional regression methods, we'd be hard pressed to hold onto everything, but we'll be taking a different approach one way or another.

As for those 10 measurements, things like symmetry and especially fractal dimension have comparatively lower correlation values, struggling to break past 0.4. On the other hand, number of Concave Points and Concavity are highly correlated with values from 0.65 to almost 0.8. The previously mentioned area/perimeter/radius trio is also in this range, but those are almost certainly highly correlated with one another.

To unpack this a bit further, let's look at a correlation heat map or two.

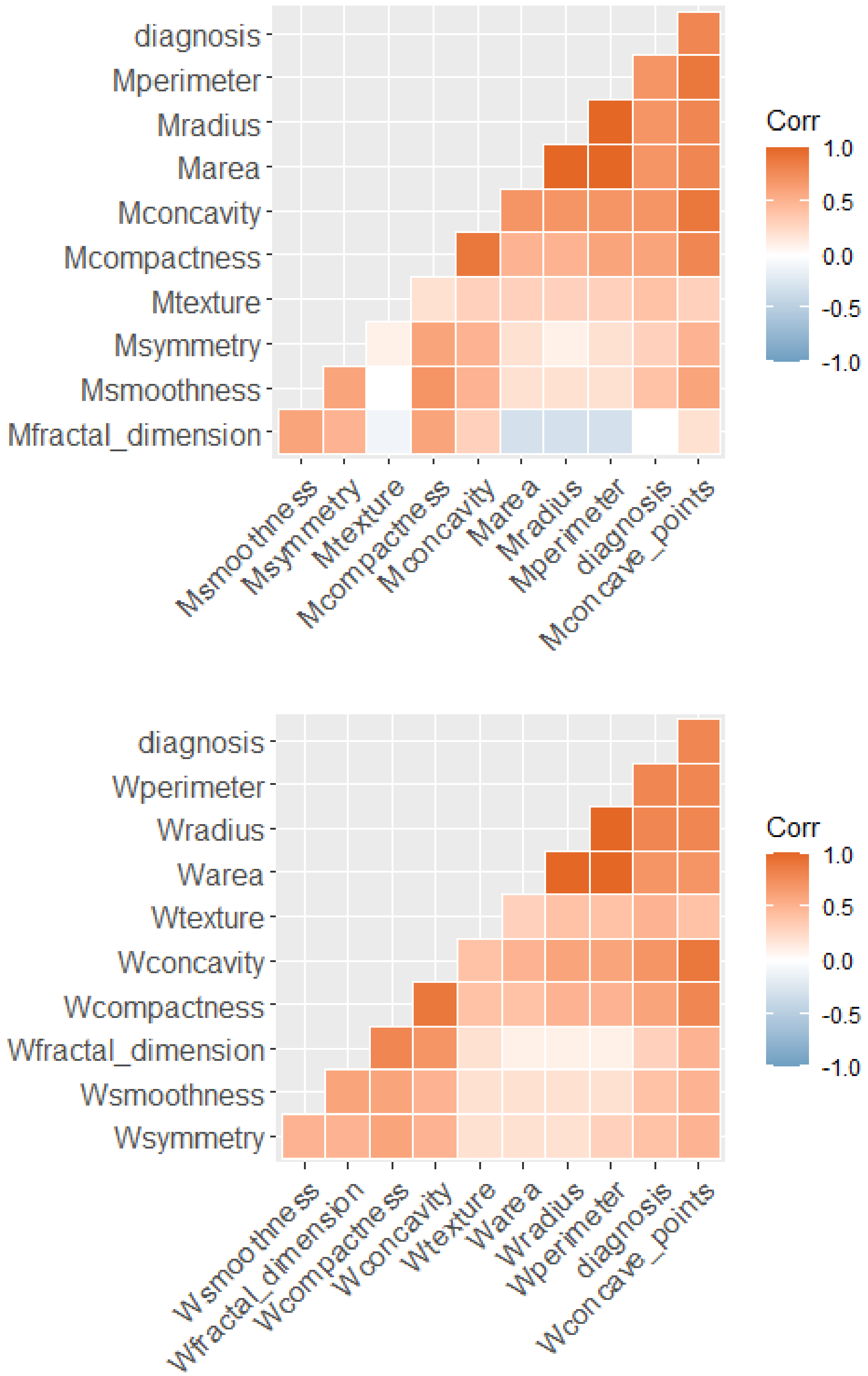

Figure 2a and 2b - Correlation Heat Maps for the Averages (1a, first) and Worst Cases (1b, second) for each measurement. The deeper the orange, the higher the positive correlation. The few sparse blue squares denote faint negative correlations.

While the are some minor ordering differences here and there, we see much of the same trends across both of these heat maps.

Much of the same we've examined previously - as well as assumed - is apparent here. concavity, concave points, perimeter, radius, and area have greater correlations with malignant diagnoses, while things like fractal dimension, smoothness, symmetry and texture have far less bearing on the matter. That first group however, and not just perimeter/radius/area, but also the two concavity measurements, are *highly* correlated with one another. The remaining measures fail to demonstrate nearly as strong correlations in just about any way, with the exception of compactness having notable correlation with the concavity measurements.

There's a few other minor points of interest as well, but only minor. Of what little negative correlation exists, the most distinct pocket of it is fractal dimension with area/perimeter/radius - those values are smaller when the fractal dimension is larger. Context is difficult to establish when "fractal dimension" is such a complex concept, but the consistently low correlation makes it less of a concern. That, and while the two heat maps are very similar, one of the biggest differences is that the correlation values are generally slightly higher across the board for worst cases over average values. Could have something to do with maximum values sort of artificially inflating the comparisons.

But at this point, the well is beginning to run dry beyond intuition, unless we can reliably address collinearity. Thankfully, there is an approach for this - Principal Component Analysis. After properly scaling our data, the eigenvalues and eigenvectors of the data's covariance matrix provide us with completely uncorrelated Principal Components to observe in a different context from the raw data.

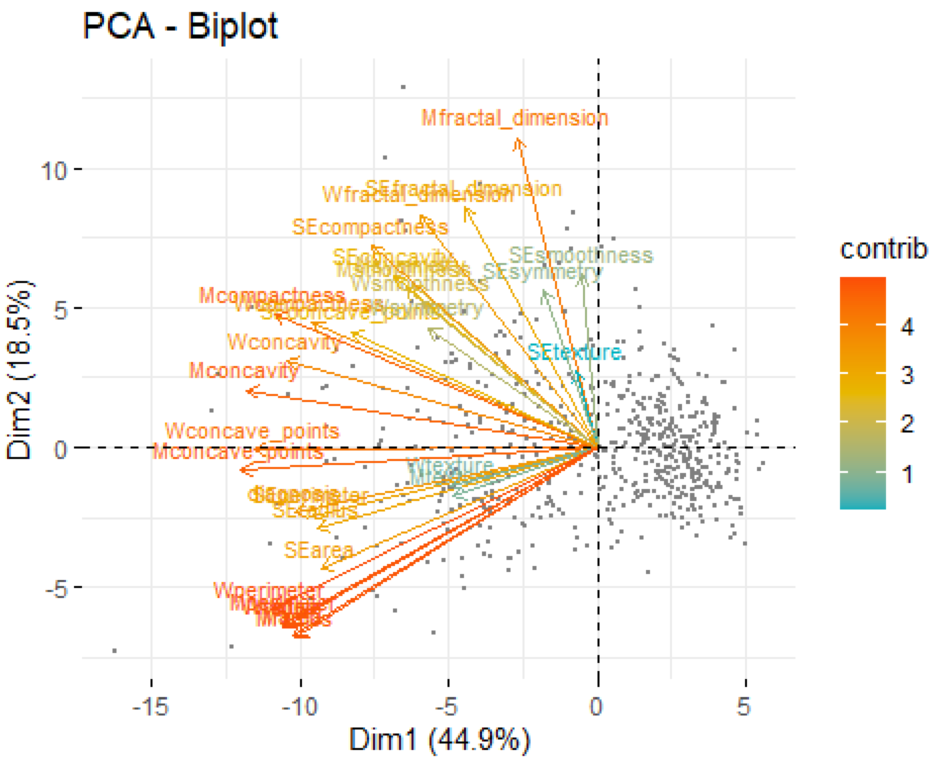

Figure 3 - A Principal Component Analysis Biplot. The first two principal components are represented here, which cumulatively account for 63.4% of the variance (not bad! just shy of two thirds). Hue of the loading arrows for each variable denote its degree of contribution. Axes are unconstrained, which is bound to result in legibility issues with thirty features.

We see a lot of vectors of high significance for variables we expect to see, and vice versa, although fractal dimension seems to have a strong presence for principal component 2, and compactness for principal component 1.

Helpful at a glance, and good to keep on hand to see all feature directions, though to really get a clearer look at what's happening, let's zoom in a bit more so it's less cluttered.

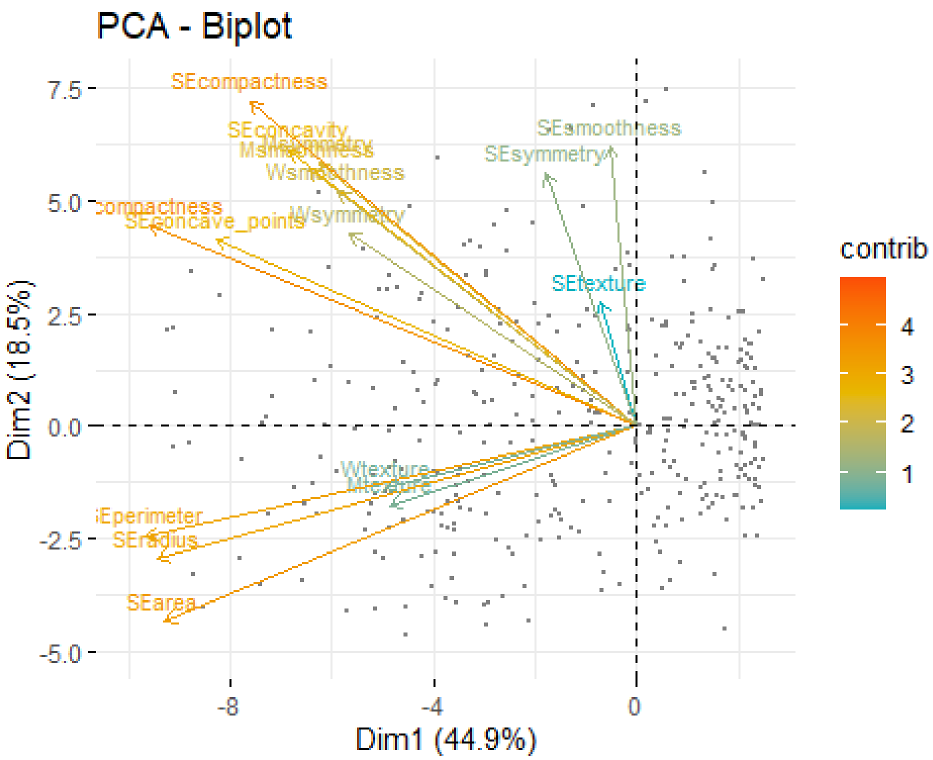

Figure 4 - A Principal Component Analysis Biplot similar to Figure 2, but with a tighter range on the x and y axes, allowing us to parse a few of the more subdued elements with easier visual clarity.

Better. We can see Standard Error elements not pertaining to our highlight factors contributing very little (and having more of a connection to PC2), texture being generally less relevant but having a stronger connection to PC1 than 2, and smoothness/symmetry behaving similarly - though in the opposite direction of texture.

This is information that could be used to more thoroughly explore the nature of our principal components, which we could do more of in the future, however, this does on its own strengthen our previously established understanding of the relevancy of our factors. We'll shelf this for this current instance of the study, however, as it's time to work on an effective model.

Modelling Methodology

Ultimately, our goal is one of accurate classification. We'd like FNAs and any other methodology that might obtain the data an FNA does to be as accurate as possible in quickly identifying cancer in a patient. Classical linear regression and its derivatives, even the more lenient ones assumption wise, are swamped in unreasonable assumptions about the nature and structure of our data. We have just enough features that our most accurate yields would likely be difficult to interpret, so we can take methodology that doesn't prioritize it in stride. Given the data volume and computational tools at hand, intensive models are difficult to justify (for example, any neural networks should be fairly simple in structure.)

To start, we will look at K-Nearest Neighbors, which does not demand high data volume or computational power, is effective with purely numerical input, is not super interpretable but need not be, and is not reliant on restrictive data distribution assumptions. We'll run a 5-fold Cross Validated sequence of this, where a specific 20% of the data set will be used to test the KNN model built around the remaining 80% - after standardizing the data of course.

This algorithm will, for each iteration, introduce observations from the validation data, calculate Euclidean distance between our standardized observations, and classify the point as benign or malignant by majority of the k number of closest training data points. We'll obtain from each iteration the accuracy of this classification and observe as well the aggregated average accuracy across all folds. Through some rigor, we'll find which value for k yields our best results. *Do we need an in-depth description of algorithms/models here?*

Given our understanding of the data and our needs from a model, there are a few other strong choices for models we'll develop to compare with KNN and see what performs the best. Naive Bayes operates on an aptly named assumption of our 30 features being independent, which we know is false, though the conceit of a joint probability distribution function of dozens of features is its own naivete *Should this last remark be kept?*. This method is known to work surprisingly well despite the assumption, especially with "high dimensional data." While thirty features is nothing to scoff at, it's also not in the hundreds, so expectations aren't too high.

A third option worth exploring is the Support Vector Machine. This will draw a hyperplane decision boundary through our thirty dimensional space and find the best one that yields the highest average distance from all the observations. While conceptually daunting and difficult to parse at a glance once you encroach even more than 3 features, this method is known to handle high dimensionality very well. Even further, this hyperplane need not be linear, and it uses dot products of vectors rather than vectors themselves, which keeps the computations from becoming alarmingly complex. We can try a number of different types of structures and see how they behave.

Lastly, a neural network model. A digital mimicry of our understanding of human neural processing, this will take our input and feed it into a few layers of processing components to parse the likelihood of each outcome. While they can be deeply powerful, the computation power they demand can get extreme. With just two hidden layers, we ran a few different combinations of "neurons" per layer, and increasing the count for both improved accuracy considerably, though the computation time became noticeably longer as well. We'll elaborate a bit later, but this is a potentially *very* strong method, if the computational power is available, and/or the data isn't prohibitively large.

Last order of business before comparing the different models is comparing the same models with different tuning parameters to find our best versions of each model. For naive bayes, there was not much to go over, and the performance of nonlinear kernels for support vector machines was so far behind the linear one they're hardly worth mentioning. We have a bit more to talk about with K nearest neighbors and neural networks.

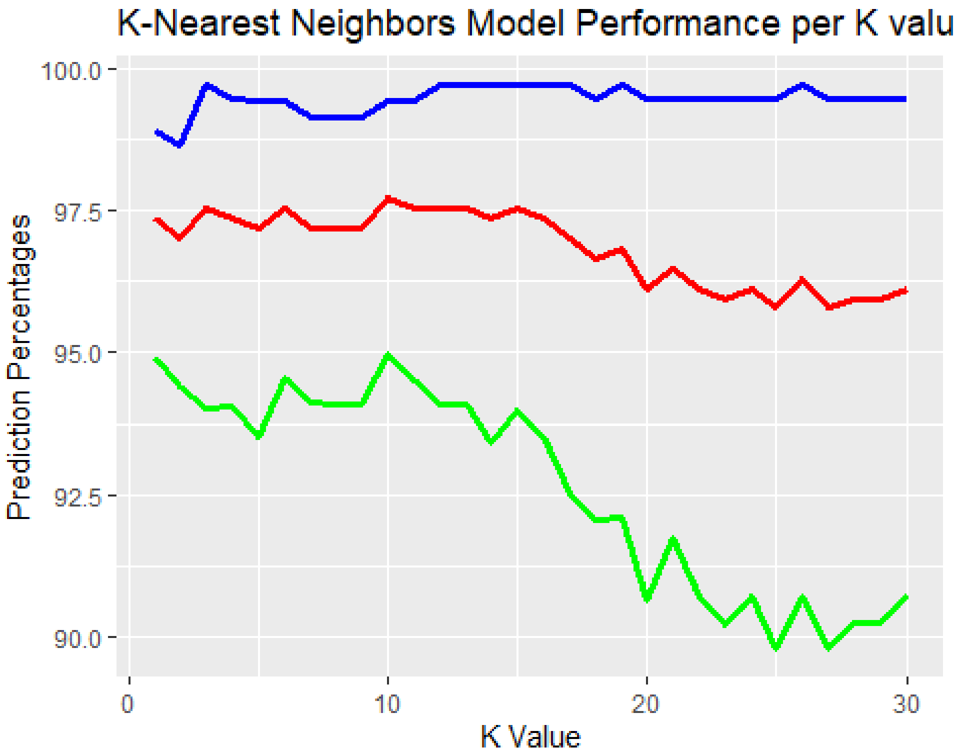

Figure 5 - A multiple line plot of our KNN models' performance across the 5 folds of our cross validated data, using all integer values 1 through 30 for K. From Top to Bottom (Blue, Red, Green) is Sensitivity, Accuracy, and Specificity.

Originally, the plan was to pick a model based on accuracy, which has us settling here at k = 10. Some smaller values of k appear competitive, especially when accounting for sensitivity and specificity, Though 10 appears to demonstrate an average best of the 3 worlds. k = 2, for example, has peak specificity but the sensitivity suffers, and vice versa for k = 3.

A couple of interesting trends are also visible here - for one, sensitivity is consistently the highest value throughout, then accuracy, then specificity. While having their own local peaks and valleys, accuracy also demonstrates a negative trend as k creeps upward compared to sensitivity, and specificity even moreso.

We'll go with k = 10, though there is a meaningful argument to be made here for opting with k = 2 if specificity is a higher priority, or k = 3 if sensitivity is a higher priority.

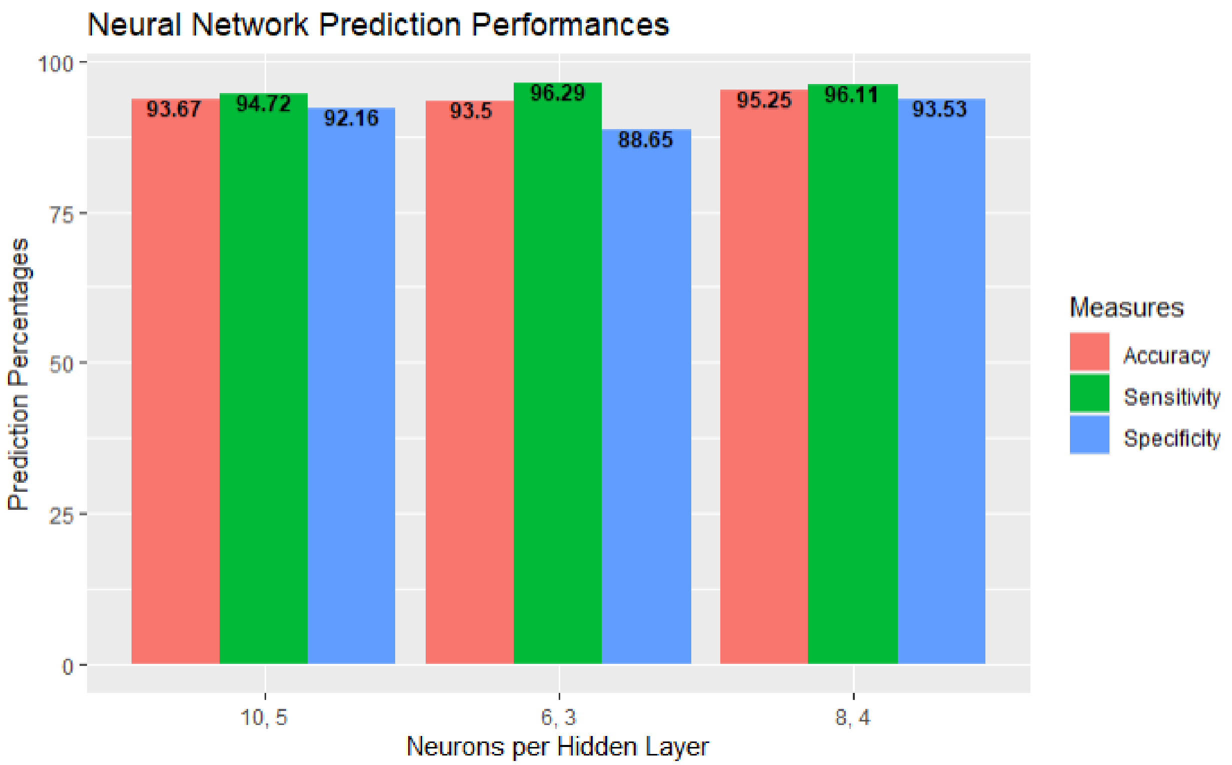

Figure 6 - A few different neural network models' performance metrics. All of these neural networks have two hidden layers - the x-axis labels refer to the number of neurons for the first, then second hidden layer.

Not too many neural network models were attempted, in part due to computation times creeping up. We tried a few different models with two hidden layers, tweaking the neuron counts, and our best results were with 8 neurons in the first, and 4 in the second. A simpler model with a few less neurons is marginally more sensitive, however in that model accuracy and especially specificity suffers - a trade-off not unlike some of the options we observed with our K nearest neighbors models. With considerably more time and computing power to allocate, it is likely that a stronger neural network model can be identified, though of the ones we tried, 8 and 4 neurons was the strongest.

Results

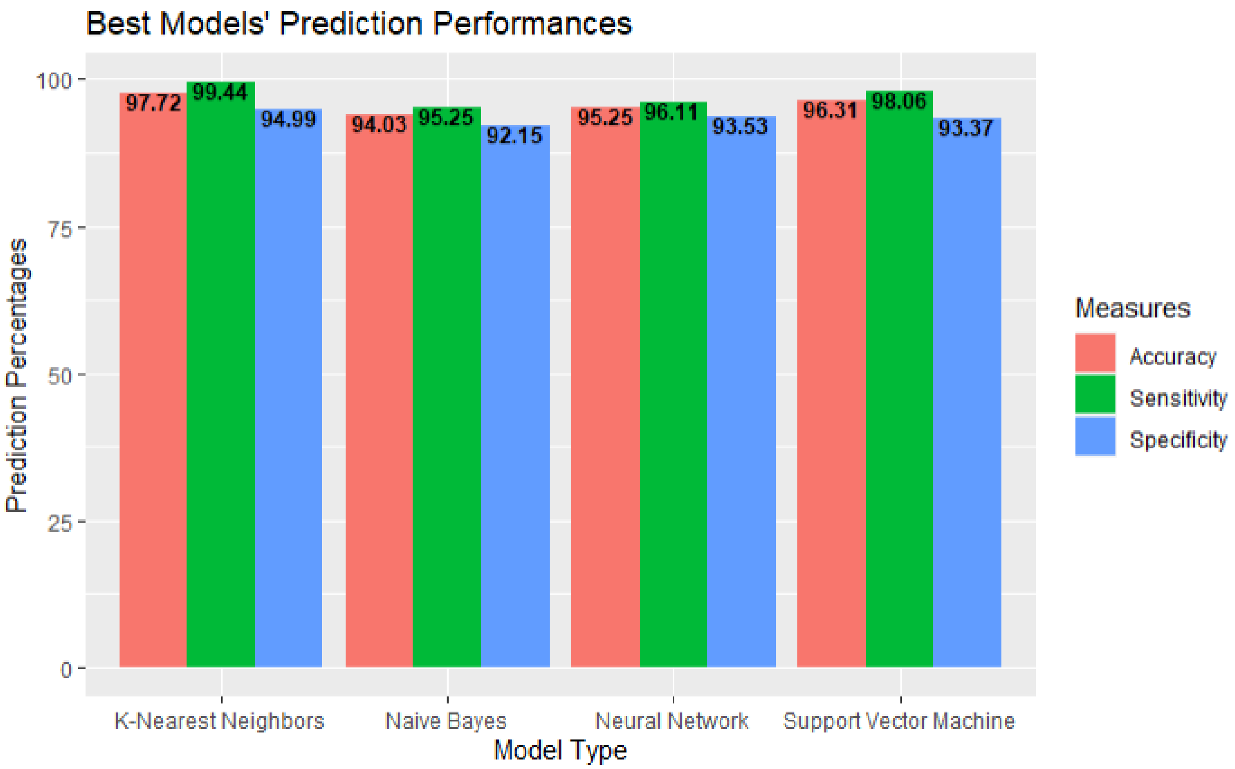

Figure 7 - The best versions of our four models and their performance metrics.

From the models we've developed, K-nearest neighbors performed the best. The Naive Bayes model, while performing respectably, was the weakest out of the 4, including another model that did not require data scaling (the support vector machine model). The neural network model had a comparatively middling performance, only outdoing Naive Bayes. However, with how flexible the structure of a neural network can be, it is very possible that a more complex setup could even outperform KNN, with the important caveat that computational power can become a bottleneck, especially if you were to entertain this modeling endeavor with additional/expanded data. The support vector machine performed fairly well though being outdone by KNN - this was with a linear kernel, as alternative, higher dimension kernels actually behaved abysmally, hovering around 62% accuracy or less. Perhaps with scaled data or with other accommodations it could be fleshed out, but neural networks has a lot of straightforward potential (though stressing computational power) and KNN still beats it outright.

Curiously, from what strongest models we've settled on, originally from the interest of accuracy, this applies almost entirely the same to all three performance metrics here. Other than the neural network having a very marginally improved specificity rate compared to the support vector machine, KNN > SVM > NN > NB for all three metrics. While not entirely impossible that an even more rigorous exploration of model types could find models that maybe sacrifice sensitivity for specificity or something else along those lines, it looks like the models we favor are the same regardless of the kinds of errors we're most interested in preventing.

Conclusion

Across four different modelling methods, all appear fairly competent at accurately predicting malignancy, though two stand out over the others, with K-Nearest Neighbors significantly outperforming our best attempts at the others with nearly 98% accuracy, and neural networks not lagging behind the most with the most malleability. If computational hardware on hand permits it, exploring more structures for neural network models could yield the best results. Otherwise, lightly tuning a K-Nearest Neighbors model appears the most reliable and strongest method out of the four. This continues to be true for the purpose of specifically preventing as many false positives specifically as possible, as well as false negatives. Unfortunately, these models are slightly more inclined to provide false negatives than false positives.

To recognize a few assumptions this study has taken place under, limited context on the data was provided. Reassuringly, For our 569 patients from Wisconsin, the model did fantastic even with cross validation, and the data is on tissue, so even if this data set was not a strong representation of the general public, it's not clear how much the model would falter with data from a more generalized pool of patients, or patients from a different set of subgroups compared to that of whatever the Wisconsin data set may have had. It is nevertheless important to recognize the finite scope of the process, as it is still possible that the data set came from some kind of skewed sub population. Further data collection and rigorous data sampling/bootstrapping always has the potential to improve and reinforce results. This was also conducted under the assumption is that there is meaningful time to get FNA data like this before a diagnosis, which is less meaningful the less time there is between the two being available, and conversely more valuable if methods of collecting FNA data can be sped up safely.

References

-Dua, D. and Graff, C. (2019). UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science.

-W.N. Street, W.H. Wolberg and O.L. Mangasarian. Nuclear feature extraction for breast tumor diagnosis. IS&T/SPIE 1993 International Symposium on Electronic Imaging: Science and Technology, volume 1905, pages 861-870, San Jose, CA, 1993.

-Kshitiz Sirohi. "K-Nearest Neighbors Algorithm with Examples in R (Simply Explained Knn)." Medium, Towards Data Science, 30 Dec. 2018, towardsdatascience.com/k-nearest-neighbors-algorithm-with-examples-in-r-simply-explained-knn-1f2c88da405c.

-finnstats. "Naive Bayes Classification in R | R-Bloggers." R-Bloggers, 9 Apr. 2021, www.r-bloggers.com/2021/04/naive-bayes-classification-in-r/.

-The American Cancer Society medical and editorial content team. “Fine Needle Aspiration (FNA) of the Breast.” Cancer.org, American Cancer Society, 2019, www.cancer.org/cancer/breast-cancer/screening-tests-and-early-detection/breast-biopsy/fine-needle-aspiration-biopsy-of-the-breast.html.