Module 5 - Diagnostic Systems

Group: 12
Anton Claesson, 971104-3217, MPDSC, canton@student.chalmers.se 17~h
Lukas Martinsson : 980203-9678, MPDSC, malukas@student.chalmers.se, 17~h

We hereby declare that we have both actively participated in solving every exercise. All solutions are entirely our own work, without having taken part of other solutions

Reading and reflection

Machine learning techniques to diagnose breast cancer from image-processed nuclear features of fine needle aspirates

Fine needle aspirates (FNA) for breast cancer screening involves taking a cell sample and spreading it on a glass slide. Digital images are then produced by capturing images of these slides.

The paper describes how a graphical user interface was created to allow an operator to roughly trace an outline of a number of cell nuclei in a digitized image sample. These outlines were refined algorithmically to produce “snake” outlines. From the snakes, 10 different hand crafted features (and their corresponding mean and standard deviation) were calculated. Most of these features were related to the shape or texture of the cell nucleus.

A MSM-Tree was used as a classifier. This classifier fits a number of separating planes in feature space. The best single-plane classifier used “mean texture”, “worst area” and “worst smoothness” only as nuclear features. Each sample point was then projected onto the normal line of the separating plane, which allowed the distribution curves of malignant and benign points to be calculated in this feature space.

A new point could then be classified by projecting it on the same normal line and comparing the distribution curves of malignant and benign points at that point.

One of the key benefits of a machine learning model for breast cancer diagnosis is that the evaluation is objective. This can be compared to diagnosis by humans which is subjective. For example different physicians will give different diagnoses on the same subject while a machine learning model could always give the same prediction if given the same data.

Additional notes:

- 569 patients (212 cancer 357 benign) for developing and 54 consecutive test patients.
- The prospective accuracy (10-fold cross validation accuracy) was 97% and the test accuracy on new patients was 100%

- This method does require a fair bit of operator input for processing the samples.

The Mythos of Model Interpretability

Key questions for a developed machine learning model still need to be asked; can we trust the model, and will it work in deployment? Is there anything else we can learn from the model? The task of model interpretation seems underspecified, what is interpretability?

Interpretability is important for a few reasons. First and foremost, interpretability is key for consequential decisions such as medical diagnosis, where one might require a reason and an explanation of a decision. Interpretability of a model might be beneficial for many other reasons as well. These include:

- If the model is interpretable, it could also be easier for a human to trust.
- Determining causality and not just correlation.
- Determining how well the model generalizes at a greater depth than just evaluating it on the performance gap between training and test data.
- Provide useful information for human decision making.
- Assessing if an algorithm’s decision making is fair and ethical.

There seems to be two main methods for incorporating interpretability in a machine learning system: using transparent model designs or making post hoc interpretations.

Transparency is a concept that indicates it is possible to understand the mechanism by which a certain model works and how and why it therefore produces a certain output. A transparent model needs to have good simulatability (it should be simple enough for a person to comprehend), good decomposability (each part of the model has an intuitive explanation) and good algorithmic transparency (there is a good understanding of the learning algorithm, for example one might then be able to know that the model will generalize well to new and unseen data sets)

Post hoc interpretability is a different approach in which the focus lies within extracting information from a learned model rather than investigating precisely how the model works. Common ways of doing this is to include natural language explanations, visualizing learned latent space representations or high-dimensional distributed representations and through explanation by examples.

Implementation

After loading the dataset we create a train-test split of 30% such that we obtain 398 train samples and 171 test samples. Then, we fit a scaler using the training data and standardize the features of the train and test data to zero mean and unit variance. As is common with diagnostic systems, we choose to measure the performance of our models by calculating sensitivity, specificity and accuracy.

Rule based classifier

We were supposed to implement a rule based classifier that could, based on the cell's size, shape, texture, or similarity/homogeneity determine if the cell is malignant. Hence we used the Eickhoff summary in tandem with the mean and standard deviation of each feature to determine which features to use and at what cutoff point each feature should indicate that a cell is malignant or not. After determining the features, the true positives and false positives for different standard deviations was analyzed to find a good tradeoff between them. To reduce overfitting we only used integer values for the standard deviation at different cutoff rates. Although the features are different, a cutoff rate with the mean + three standard deviations off tended to give a good tradeoff between true positives and false positives and was therefore used for all of the features utilized. Listed is the training accuracy for each feature used on their own:

Feature	Accuracy
area_0	83.4%
concave points_0	86.2%
texture_0	62.6%
perimiter_0	81.2%

Using the rule based classifier we obtained the following results on the train/test sets respectively:

Train metrics: [Accuracy: 0.892, Sensitivity: 0.765, Specificity: 0.968]
Test metrics: [Accuracy: 0.901, Sensitivity: 0.746, Specificity: 0.991]

Random forest classifier

Using Sklearn we perform an exhaustive search over a few different sets of hyperparameters of the random forest. In particular, we vary the parameter “n_estimators” between [3, 5, 10, 20, 40, 60, 80, 100] and “max_depth” between [1, 2, 3, 4, 6, 8, 10].
For each parameter combination, we perform 10-fold cross validation on the training set, picking the combination that yields the highest scoring sensitivity (recall).
We pick sensitivity as a metric as we noticed this metric is the one the model has the most trouble optimizing for. Additionally, having a high sensitivity could be argued to be especially important in the case of cancer diagnostics as we don’t want to incorrectly label a positive sample.

The resulting best parameters are the combination of “max_depth”= 10 and ” n_estimators” = 60. We then re-fit a random forest classifier on the full training set using these parameters, and obtain the following results when predicting on the train and test data respectively:

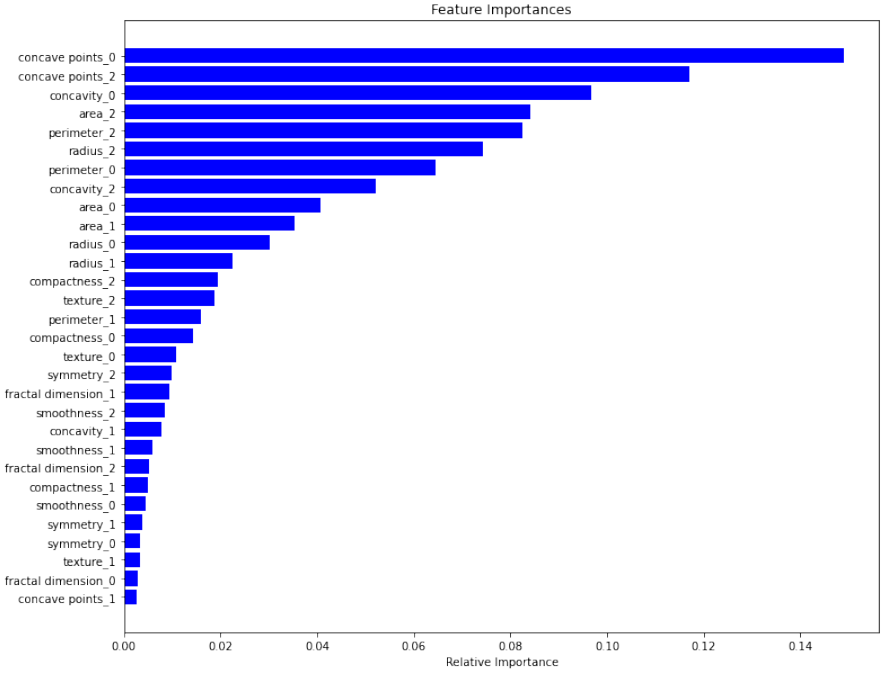
Train metrics: [Accuracy: 1.000, Sensitivity: 1.000, Specificity: 1.000]
Test metrics: [Accuracy: 0.971, Sensitivity: 0.937, Specificity: 0.991]

As we can see even though we have used 10-fold cross validation it seems as if our model is overfitting to our training and validation data. If we didn’t have a test set and we went only with the data from the 10-fold cross validation, we might incorrectly assume the model would work perfectly. This shows the importance of using a test set even though cross validation is applied.

Custom classifier (logistic regression)

One problem with our model in terms of interpretability is that we are using quite many features. In most cases, the features are of varying importance when it comes to predictive capability. Luckily, instead of doing manual feature selection we might use our random forests to learn feature importance.
Plotting these in Fig. 1, we see that the most important features seem to be related to “concave points” and “concavity”, i.e. features that measure the smoothness and irregularities of cells.

Fig 1: Random forest feature importances

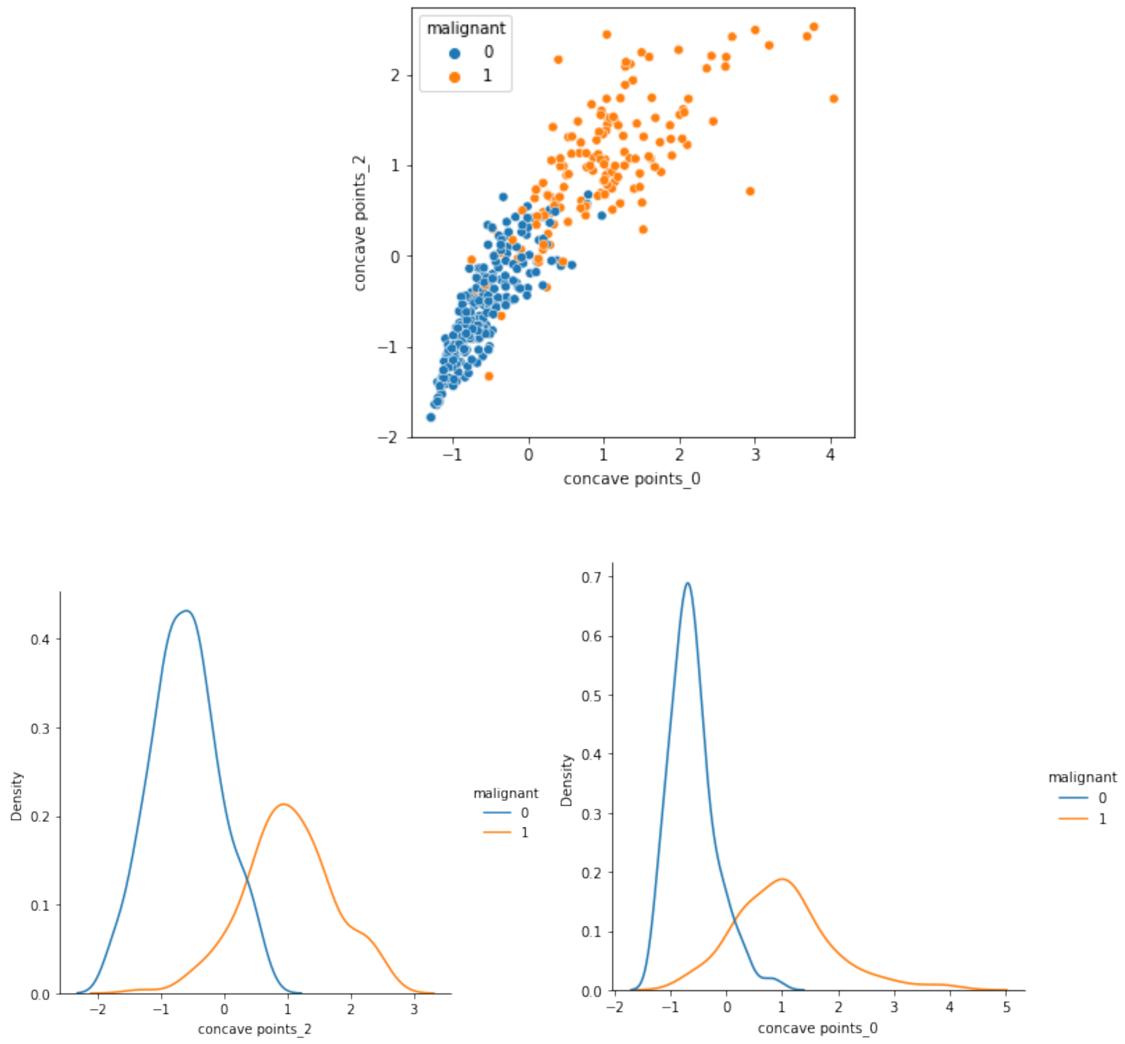


Now, we argue that in this case interpretability could be very important. We want to see why our classifier is classifying a certain sample as either malignant or benign, and we want to understand why. Additionally, from the feature importance plot we see that we might be able to obtain an alright performance using only the two most important features; “concave points_0” and “concave points_2”, the mean and the variance of concave points.

Selecting only these two features would also surely help us avoid the problem of overfitting, but as this might be a bit extreme we would of course expect quite a significant drop in performance.
However, when we have only two features we can exactly visualize the feature space and our model’s decision boundary, which can be argued is a huge plus in terms of interpretability.

In Fig. 2 the feature space is visualized along with the respective distribution of malignant and benign samples for each feature respectively. We see that while a linear model won't be able to separate the two classes fully, it would still perform quite alright given that samples in the upper right seem to be malignant and samples in the lower left seem to be benign in general.
We therefore select a Logistic Regression model as this is the simplest linear model which will attempt to separate the clusters by a line.
Moreover, from the plots we see that the selected features are obviously very good, indicating that the random forest did a good job at ranking the feature importances.

Fig. 2: Feature space and distributions

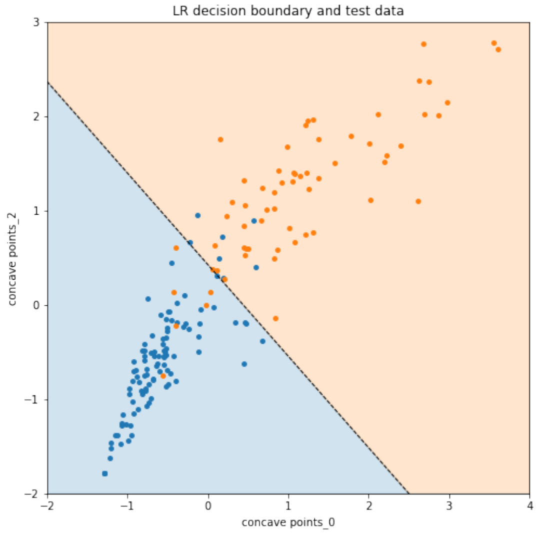


We fit a logistic regression model to the training set with the default parameters provided by sklearn, which adds an L2 regularization term to the loss function which reduces overfitting. However since we only have two features in this case this would not be necessary. We obtain the following results:

Train metrics: [Accuracy: 0.927, Sensitivity: 0.872, Specificity: 0.960]
Test metrics: [Accuracy: 0.924, Sensitivity: 0.905, Specificity: 0.935]

As we hypothesized we no longer have overfitting but we have a significant drop in performance, most in terms of accuracy and specificity, in comparison to the random forest. However we are now also able to plot the exact decision boundary of our model and compare it to the test set data points, which is shown in Fig. 3. This allows us to also visually predict what a new sample would be classified as.

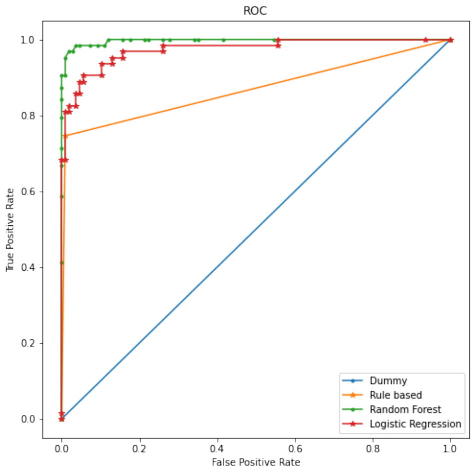
Fig 3: Logistic Regression model decision boundary and test data points.



ROC Comparison of model performances

An ROC curve allows us to see how varying the discrimination threshold in a binary classification problem impacts the true positive rate and the false positive rate, which is very relevant when it comes to diagnostic systems. We therefore choose to also plot these for the three models we have constructed, and a dummy classifier which always predicts “malignant”. These are shown in Fig. 4.

Fig 4: ROC curves of the different models.



Note that the rule-based classifier always has either a probability of 0 or 1, so the curve looks a little bit weird. Regardless, analyzing the area under the curve we see that the random forest provides the best trade-off between true positive rate and false positive rate, followed by logistic regression, rule based and finally the dummy classifier.

Discussion: Anton

From this week's lectures and assignments we have learned that interpretability might be just as important as the predictive capability of machine learning models depending on setting. As the paper discusses, what an interpretable machine learning model actually is is oftentimes quite vague and undefined. However, when it comes to diagnostic systems I think it is very important that the model can be trusted, and that is something that the paper also touches upon. If one understands how the system is making a decision one might be more inclined to trust it. Moreover, transparency is another thing that is discussed and I agree is very important in this setting. I think trust and transparency might be quite closely tied to each other but let us discuss an example.

If we again take a look at the decision boundary of our logistic classifier we might say that we would be able to trust the model since we know why and when it is going to make a certain prediction. This model is also very transparent; if we observe a sample close to the decision boundary we might conclude that maybe it is wise to do another checkup, but if we are very far from it we might be able to trust the decision fully. Of course this could in more complex cases be expressed in a numerical form in terms of probabilities, but I think this visual representation increases human understanding, transparency and thus trust of the system.

In this example we also experienced the trade off between interpretability and performance, as when we use the selected 2D feature space the samples are very obviously not separable. It might be that the samples could be fully separable in a higher dimensional space, but that would not allow us to see the exact decision boundary and thus reduce the system's interpretability.

Discussion: Lukas

Interpretability can have many different meanings and explanations depending on the application and domain. When it comes to machine learning this is hard to define. This is because what it is at its core is how transparent something is, however depending on the machine learning model this is far from trivial. One could argue that models have to be interpretable to hold some sort of accountability, this is especially true in terms of medicine or self-driving cars. If a model makes a prediction that determines someone's life and at the same time cannot explain why that choice is optimal it can be difficult to argue from a moral standpoint why you should put your life into something that can be explained. However, I would argue that as long as it is capable of saving more lives, e.g. in the case of medicine it should not really matter. Then the reason of interpretability importance has more to do with keeping the models updated and as accurate as possible. The dilemma that becomes more ambiguous is when the model has a higher accuracy than a human but when it makes mistakes it is on prediction that a human would correctly predict. Therefore some hybrid where a machine points out the diagnosis and its reasoning behind it would be optimal. And in cases like with linear regression this reasoning is easy, however in more black box models this becomes rather difficult. Then the model either has to point to previous similar cases to make the human agree with its decision or the model has to be developed to explain its reasoning, which is not always possible. This phase however, with a hybrid prediction of both humans and machine learning models is only a temporary phase in my opinion. As the models become magnitudes better at predicting outputs then humans the trust in them will slowly increase. Nevertheless, interpretability in its current state has to be defined. It could be through either visuals in medicine, where the model points out where it has found for example the malignant cells or through raw numbers. I would define interpretability in the context of ML and medicine to be how well a domain expert (or a ML expert working together with a doctor) could understand why a model makes a certain choice over another. The explanation that makes them come to this understanding could vary, be it from visuals to numbers or something else.

Individual Summary and Reflection: Anton

Lecture summary (Diagnostic systems)

The lecture first covered a short history of AI diagnostics in medicine. In particular, early systems were based on “expert knowledge” to make handcrafted rules and features.Most notable problems of these systems are that they scale and generalize poorly, lacked training data and of course required human experts.

The lecture also talked about what diagnostics is, i.e. finding the cause of some symptom(s) based on tests and data. We covered the naive bayes classifier for diagnostics, which has the advantage that we can add new features with time. Furthermore, metrics such as sensitivity and specificity were explained. We also covered why AI will have a huge impact in the medical field; as an example speed of workflows can be improved and new unknown features and explanations of diseases might be uncovered from data that is becoming increasingly more abundant and standardized. Finally, we also talked about some problems such as scarce labels and that the stakes are very high since it can be the matter of life and death.

Lecture summary (NLP follow-up)

The follow-up session was divided into three parts. The first part covered the different questions asked in the assignment, and some discussion about them. For example, rule based interlingua approaches are quite similar to state-of-the-art neural systems as an intermediate representation is computed, and that rule based systems might be preferable if data is scarce.

The next part covered discussions about the decoding part of the assignment and problems encountered here.
In the third part some more “philosophical” questions of NLP systems were discussed such as how the models might mirror unfair biases in the real world, and what consequences could come from this.
Finally, some ways to evaluate translation systems were also discussed, for example by using human evaluators or comparing the translations to reference translations in an automatic way (BLEU score, i.e. overlap of n-grams).

Reflection on the previous module (NLP)

One of the key take-aways for me from this module is the fact that it is possible to construct such a different amount of various complex systems for the same task, in this case automatic translation. It’s interesting to see how solutions to an AI problem that has been around for such a long time has evolved over time.
I also found the parts about NLP stereotypes interesting, and I think it is important to consider the consequences of using real-world data and all its biases when creating AI systems. Hopefully it will be possible to combat unwanted behavior and “steer” NLP models in a desired direction in the future, and from what I have seen it is currently an area of active research.

Individual Summary and Reflection: Lukas

Lecture summary (Diagnostic systems)

The lecture covered the history of machine learning in medicine, from “simple” knowledge based models to the neural networks used in the 1990’s. It was interesting to understand how little of the data in medicine actually was standardized, even up until recently. Then specificity and sensitivity was explained (which is really interesting to know since the PCR tests used these) and how to find a tradeoff between how to demonstrate it with for example a ROC-curve. Following was an in depth discussion of when to use Naive Bayes to predict something in reverse could be more beneficial then doing it straight up. Lastly the potential benefits of ML models in medicine was explained, since it can be both faster and more accurate then any human. However, since it is not always easy to understand ML models', their “reasoning” is something that must be taken into account , especially when lives are at stake and one has to provide a moral argument for why even though they might be better than humans they can still be at fault.

Lecture summary (NLP follow-up)

The lecture was divided into three parts. First the questions posed in the assignment “Can you think of similarities between rule-based translation systems and the state-of-the-art neural systems?” and “Can you think of situations where it could be preferable to use an “old-school” rule-based solution instead of a modern?” was discussed. Solutions such as when the data is scarce or to simplify learning was brought up. The second part covered the decoding task and how to handle larger texts. Third and lastly discussion regarding stereotypes, bugs and evaluations regarding NLP was talked about. For example if a translation is in reality a feature or an unknown consequence of the system. And even if it is, is it considered a bug or not?

Reflection on the previous module (NLP)

Since I had no previous idéa on how NLP works, this module was really interesting. Especially how one can use Naive Bayes to predict both the probability of the occurrence of a word and the reverse translation to design better NLP models. Furthermore it was intereseting how one can train a all purpose NLP model and then train it further on a more specialized set that one wants predcit on (like in the paper where a model was design to precict suicide attempts based on social media). This approach thus enables models to get a higher accuracy on datasets with scares data. Lastly, the more philosophical discussion of what is considered a bug vs feature as well as how an unwanted bias can steer models to be more biased towards a certain word without humans realizing it I found to be particularly interesting. That and the difficulty of when to use he/she and if a language that has a gender neutral pronoun should be used instead of what the model normally would predict.