# Statistical testing:

1. Assume that two systems are used for a binary classification task on 100 test items, and that accuracy is calculated for each system. Derive the relationship between the accuracies and $k$ as used in the sign test under the assumption that no correction is needed for the number of ties being odd.

   Accuracy is the number of items which a system gets right divided by the number of items which the system got wrong.

   In the sign test, $k$ is given by:

   $$k = \frac{null}{2} + \min(plus, minus) \tag{1}$$

   The relation between $k$ and accuracy is that:

   $$\Delta \text{ accuracy} = 1 - \frac{2k}{n} \tag{2}$$

   Note that $2 \times k = null + 2 \times \min(plus, minus)$ and that $null + plus + minus = n$. This means that $n - 2 \times k = null + plus + minus - null - 2 \times \min(plus, minus)$. This is equal to $|plus - minus|$.

   We divide both sizes by $n$ by $n$. So $1 - \frac{2k}{n}$ is the difference in accuracy between the two systems.

   So a higher $k$ suggests that the difference between the two systems is not significant while a lower $k$ suggests that it is. For example if $k = \frac{n}{2}$; then $2 \times null + 2 * \min(plus, minus) = n$. This means that $plus = minus$ suggesting that the systems are no better than each other. This suggests there is no difference in accuracy.

   However, if $k = 0$; then $null = 0$ and $\min(plus, minus) = 0$. This suggests that one system got everything right while the other got absolutely everything wrong. This would suggest a very strong significance and a very high difference in accuracy.

   Obviously there is a linear scale between the two extremes where a lower $k$ suggests a higher difference in accuracy.

2. The number of ties found between two systems is calculated as part of the sign test. How might this information be used in designing an improved system?

   Knowing the number of ties can give us a measure of how "similar" or different the systems are. We can use this to see the magnitude of the change that the differences between the two systems made This allows us to have a better intuitive idea of the magnitude that these changes would have in a new system. If we noticed that ie whenever the two systems agreed they have a very high chance of being correct; then we could potentially use these two systems as sub-systems in a newer system to "deal with the obvious" cases.

# Overtraining and cross-validation:

1. Suppose you test a binary classification system using 10-fold cross-validation with 100 items in each fold. You obtain the following results for the folds: 81, 86, 82, 84, 79, 79, 76, 82, 85, 88. What is the mean accuracy and the variance?

   Each result was tested on 100 elements so the accuracy is the number which it got correct divided by 100.

$$\begin{aligned}
\overline{x} &= \frac{\sum_{k=0}^{n} x_k}{n} \\
&= \frac{\frac{81}{100} + \frac{86}{100} + \frac{82}{100} + \frac{84}{100} + \frac{79}{100} + \frac{79}{100} + \frac{76}{100} + \frac{82}{100} + \frac{85}{100} + \frac{88}{100}}{10} \\
&= \frac{81, 86, 82, 84, 79, 79, 76, 82, 85, 88}{1000} \\
&= \frac{822}{1000} \\
&= 0.822 \text{ to } 3.S.F.
\end{aligned} \tag{3}$$

$$\begin{aligned}
\text{Var}(x) &= \frac{1}{n}\sum_{i=0}^{n}(x_i - \overline{x})^2 \\
&= \frac{0.01196}{10} \\
&= 0.00120 \text{ to } 3.S.F
\end{aligned} \tag{4}$$

2. An alternative system, tested using exactly the same folds, gives the following results: 82, 87, 83, 85, 80, 81, 77, 83, 87, 89. Could this result be statistically significant at the 5% level? Explain your answer. (Full significance testing is not required.)

$$\begin{aligned}
\overline{y} &= \frac{1}{n}\sum_{i=0}^{n} y_i \\
&= 0.834
\end{aligned} \tag{5}$$

   Initially looking at this, you would conclude that this is not significant – since the difference is so marginal. However, on closer inspection, you should note that every data point in the second sample is (marginally) higher than every corresponding data point in the first. This means that using the sign test we would find that the probability that the second system is better than the first is $\frac{1}{2}^{10}$. This is 0.0009765625 and is highly significant at the 5% level. So we would conclude that the second system is better than the first.

3. What effects can cause the accuracy of a sentiment analysis system trained on old data using bag-of-words to decrease when applied to later data?

   The data is not the same – simply put we can only apply a model onto the data on which it was trained – if it was trained on old data then it is not useful on newer data which will be slightly different.

   Changes in both the frequency of use and style of use of words, the creation of new words and phrases. New formats of media will change how we talk Past a certain point there will also be language change.

   Over time, the sentiment of words can change –

   The creation of new words will also affect the accuracy of sentiment analysis – for example many sentiment-carrying words are very recent – the majority of grammatically neccessary words are centuries old – a reading of victorian (or older) literature will show their use has remained largely unchanged – however words conveying emotion change far, far more frequently. This means that if 3% of the words which are used are new and do not have an assigned sentiment, they are more likely to be the words that carry the strongest sentiment.

   Changes in the frequency of words will also affect the strength of their sentiment – for example a dataset trained on data from the early 2010s may find "lol" as a strongly

positive token – while if it were trained now, it would find it only weakly positive – due to overuse.

A further example would be the change in media and style of that media – ie training on emails and then applying that to modern social media – there would be a lack of similarity and so a low accuracy.

## Uncertainty and human agreement:

1. The experiment in Task 1 where you all had to choose between positive and negative for a movie review which was more accurately described as neutral sentiment demonstrates that kappa will be much higher on some items than others. However, adding a third category won't necessarily improve kappa. Why not?

   We can subdivide reviews with a neutral sentiment into two types: those which are overall very lukewarm and express no particularly strong either way ("this film is okay – I guess I'd watch it again") and those which express strong opinions in both ways which cancel out ("the special effects were spectacular! Unfortunately the characters were very 2-dimensional.").

   For the first type of neutral reviews, people will generally agree they are neutral. Adding a third category would likely increase "actual" agreement for these types of reviews (I say actual agreement because it is unlikely to actually increase kappa – I discuss this later). However for the latter type – where strong opinions are expressed both ways – it is far easier to interpret one as either positive or negative. A person who likes special effects may view the example as positive while one who cares more about character development may view it as negative – and a third person may well view it as neutral. In this case rather than increasing agreement we would likely decrease the agreement by adding a third category for people to decide the film fits into.

   This is a weakness of all measures of agreement (kappa being one of the most commonly used measures of agreement) – there are no associations between the categories and so adding any further class divides the agreement even more – take the case where people are asked to assign a score of 1 - 10 on 10 categories and then we use the distinct combinations of that as the agreement. This would mean there would be almost no agreement by any despite many of the classes being related to each other and peoples opinions being unchanged.

   I hypothesised that the interpretation of kappa is dependent on the number of categories and that a higher number of categories will decrease the kappa even if agreement is the same – this effect is because kappa does not relate any classes to any other classes: the only agreement that kappa models is when people think multiple things are in the same class – if someone rates a film 5-stars then they are just as much in disagreement with the person who rated it 1-star as the person who rated it 4-stars.

   I decided to model this hypothesis. The results of the experiment are below.

   The model I chose was that each item had an inherant value and peoples interpretations of that item were normally distributed around that value. People are then classified into $n$ equally-sized categories. This means that peoples views are related to each other and there is agreement. I ran 10 iterations with each iteration having 1000 items and 100 people.

   This simple model proved to demonstrate my point better than I had expected or hoped.

   When I simulated this model with $n = 2$, $\kappa$ was high enough to show a link – in this experiment $\overline{\kappa} = 0.55$. This showed that people tended to agree but there was not a strong concensus – this value of $\kappa$ is a plausible value.

   Introducing more categories made kappa decrease vastly. With three categories kappa

fell to 0.39, and after introducing 10 categories $\overline{\kappa}$ was 0.12 – seemingly showing that there was almost no agreement whatsoever.

The full results of my simulation are below:

| No. of categories | kappa values | mean |
|---|---|---|
| 2 | 0.544, 0.547, 0.524, 0.557, 0.538, 0.555, 0.587, 0.537, 0.547, 0.553 | 0.549 |
| 3 | 0.380, 0.394, 0.385, 0.384, 0.406, 0.405, 0.396, 0.383, 0.418, 0.384 | 0.393 |
| 10 | 0.119, 0.128, 0.110, 0.109, 0.117, 0.108, 0.115, 0.125, 0.120, 0.114 | 0.117 |

This shows that kappa is only relative and cannot be used to compare things where there are different numbers of categories. In every single model the "agreement" was identical – the mean view was unchanged and the standard deviation of that view was also unchanged. The only thing that changed were the number of categories. This simple model has demonstrated one of kappas many flaws.

The other two major flaws with $\kappa$ are that it is dependent on "trait prevelance" – if the categories have a uniform distribution then the chance agreement is lower. This means that $\kappa$ for different distributions which have the same agreement but different trait distributions can have very different $\kappa$. In short $\kappa$ is also dependent on the distribution of the traits.

Since we are now introducing a third category – neutral, most likely the classes will no longer be approximately equal. This means that the probability agremeent due to chance will be higher compared to my model, decreasing the $\kappa$ even further. This suggests that the $\kappa$ fall even faster than my model predicted (since the categories are not all equally likely).

Furthermore, $\kappa$ is also highly dependent on the probabilities of unlikely classes – so-called "marginal homogeneity". This means that for exmple, if neutral was to be a pretty unlikely class, it would still have a significant affect on $\kappa$ even if it was only to change peoples opinions from the less likely view.

Overall, we can conclude that $\kappa$ is a highly flawed metric and although whether adding a neutral class would increase the "real" agreement, kappa would almost certainly decrease it's estimate of the agreement.

2. Why might it be informative/useful to use human annotation on a sample of data even if you already have annotation which corresponds closely to ground truth, such as movie review stars?

   We would be able to use this human annotation to remove examples of more contraversial data from the dataset – for example if we have a review which is 3-stars and the human reviewers disagree – why should we arbitrarily set it to be either positive or negative? Doing such would "dilute" the truly informative data we have with arbitrary randomness and reduce the quality of our system.

   There may also be some cases where the sentiment / category is very debatable if at all applicable: for example if we were asked to classify (unbiased) news stories into positive or negative sentiment we would struggle since there is no inherant sentiment – in this case human agreement would be very low and the "gold" standard may not be truly correct or perhaps even arbitrary.

   Using human annotation on a sample of data can give a ground target for what a "reasonable" target of accuracy is – for example if we had a metric that people overwhelmingly agreed on then we would not be content with a system that matched the gold-standard only 40% of the time. However, if we discovered that humans got the prediction incorrect 80% of the time (say it was a contraversial topic or there were many categories) then we would be satisfied with a system that achieved 40% accuracy.

# 1 2021 Paper 3 Question 9

https://www.cl.cam.ac.uk/teaching/exams/pastpapers/y2021p3q9.pdf

You work at a social media company, and your task is to detect cyberbullying messages based on the text they contain. You have access to a large number of messages, N, which have been manually labelled as "OK" and "bullying".

(a) How can you apply a Naive Bayes classifier to the task and evaluate it? Describe the approach, including how you would estimate the parameters.

You would apply a Naïve Bayes classifier to the task by creating two categories "OK" and "Bullying". You would then work out the probabilities that an object in each category had the features that we have observed (in this case each feature would be the presence or absencce of a given word).

We work out the logarithmic probabilities of a given class, and for each word, the logarithmic probabilities of that word being in the class. We then work out the probability of observing all the features that we have observed given that the object is in a given category. The category that we decide the object belongs to is the category which we estimate has the highest probability.

We work out the probability of seeing a feature in a given category by counting the number of examples in that category which contain the feature and divide it by the number of features observed in all the examples in that category (each smoothed by adding one).

This gives the formula

$$\hat{\mathbb{P}}(w_i|c) = \frac{count(w_i, c) + 1}{(\sum_{w \in V} count(w, c)) + |V|} \tag{6}$$

In the example of bullying the features would be the words which occurred – however we should only count each word once since, as discussed last supervision, modelling the data as a "set-of-words" rather than as a "bag-of-words" often yields a better system.

(b) You decide to use precision and recall instead of accuracy as the evaluation metric for this task. Why does this decision make sense, and how are the metrics calculated?

The majority of examples are not bullying – and so we can achieve a very high accuracy without predicting anything being positive. For example say that 0.1% of examples are bullying: if we have a system that always predicts that messages are "OK" then we will end up with a 99.9% accuracy – despite the classifier never predicting that any messages are "bullying" and so being totally useless.

Precision is a measure of the proportion of positives in a system which are true positives. This would be used if the social media company wished to ensure that it was not "censoring" people. However, it is still possible to get a very high precision by always predicting that things are "okay".

Recall is a measure of the proportion of negatives which are true negatives. This is obtained by dividing the number of true negatives by the total number of negatives. However, this can be skewed by saying that every system is positive (and so having zero or near-zero) negatives. Using recall as a metric will favour having a very low proportion of false-negatives. This would be good if the social media company wanted to have a zero-tolerance policy on bullying.

However, both precision and recall have their flaws – it would be better to use the F-measure as a metric. This is a weighted geometric average of both precision and recall and takes a parameter $\beta$ which you use to indicate which is more important. Setting $\beta$ to be high favours recall (low false-positives) while a low $\beta$ favours precision (low false-negatives).

(c) Your colleague wants to hire two more human annotators to re-label your training data. Why might this be a good idea, how would you measure agreement in this task, and do you think this would improve your classifier in any way?

- You want to know how good the "gold" standards are – ie remove mislabelings.
- You can correct errors in labelling?
- You want to remove contraversial data from the dataset so you train your data on the best data.
- You want to update the system – but in this case why would one relabel old data rather than make a newer one?

You would measure agreement using either $\kappa$ or the agreement coefficient etc.

The main reason for re-labelling data would be to remove errors in the "gold" standard – given that there is often no inherant indicator of the sentiment of posts – and when there is often there is some user-error. For example say that we were to use user-reported bullying messages as the "bullying" dataset and arbitrarily select other messages for the other dataset. This sounds reasonable and is likely what would be done in real applications. However: there is the chance that some data in the "OK" dataset would by chance be bullying. This would dilute the true data and end up giving a worse system. There is also the converse side where messages reported as bullying may be misclicks or jokes or it may be decided that the message is otherwise not as offensive as the reporter believed. In any case; there is no true "gold" standard in this. So it makes sense to have the two humans review and relabel the data. If either disagree with the gold standard then we should not use the data in our dataset as it would skew the predictions.

We could also measure the relabellers agreement with each other and calculate $\kappa$ to see how much they typically agree. we could furthermore

(d) Due to repeated media coverage of cyberbullying, your company introduces a new policy stating that as many cyberbullying messages as possible are to be found and deleted, while still making sure the number of non-filtered messages remains high. Some additional manual labour is made available for this change. How does this affect your evaluation strategy, and how can you adapt the classifier to comply better with the new strategy? (Tip: a development corpus could be of use.)

The originl strategy involved filtering out the worst cyberbullying messages while attempting not to "censor" any which are not related to cyberbullying. Now that strategy has changed to "catch all the cyberbullying messages" and then manually filter through them using humans. This has shifted the evaluation method from a F-measure with a high $\beta$ to one with a low $\beta$ (changing from weighting precision higher to weighting recall higher).

In order to tune our system we must finetune parameters – for example the cutoff point. We need to be able to test this on realistic data and so should use a development corpus to enable us to finetune the parameters to maximise the proportion of cyberbullying cases which we catch while also keeping the proportion of cases which we catch at a manageable level such that we have the ability to manually filter through them.

(e) You realise that in this particular application, language change might cause the performance of your trained classifier to drop considerably over time. You have some manual labour available to address the problem, but not enough to relabel large amounts of text.

(i) Why is language change relevant here, and how might you notice it?

We are training data on a social media network – and on social media, language change happenns very regularly. Alternative spellings and abbreviations become

mainstream, trends come and go – with them some very specific words that will be used a lot and potentially have a high intensity. If we do not tailor our system to this then it will be significantly worse after a few years (or even months). For example if we trained our system on data from 2014 then it may find "Ice-Bucket Challenge" to be a common phrase with a strong positive sentiment. However, within a year or so this is not true and the impact our system places on the phrase "Ice-Bucket Challenge" is incorrect. This brings me onto the next point: as languages change the frequency of words use changes meaning that the estimation of frequency in our system could be orders of magnitude off.

Over time, words also take on new meanings and change sentiment. Words which had a generally negative sentiment can change to positive or words which have a positive sentiment can become negative as public opinion changes. Failure to adapt the system to deal with this will lead an accuracy loss as we get false-positives and false-negatives.

There are a few ways to notice language change. I summarise them below:

- Use on online lexicon. This is the most obvious and most "lazy" approach. Someone else has already done this work so why redo it. However, this is general and not specific to the forum on which our data is.
- Keep a running track of the frequencies of words in messages which we predict as "cyberbullying". Then report to a human the words which are changing sufficiently (in absolute frequency for lower frequency words) or in percentage frequency for higher-frequency words. For example if we notice that there is an increase in the proportion of cyberbullying messages containing word $\alpha$ then we may decide that the importance of $\alpha$ should be increased and retrain the system.
- Retrain the system regularly on new data and see how the weights for words have changed compared to the system trained on older data – are there new entries – which have greatly increasaed or decreased?

(ii) How could you efficiently build a "cyberbullying lexicon" containing words that have been found in recent cyberbully incidents?

In order to find "high quality" data about cyberbullying: my solution to this would be to include messages which had been user reported – this would contain in general lots of material which users felt was unsuitable for the platform. While there is a lot of material which is not reported, this would capture a good amount of it. This would enable us to rapidly build a large dataset containing data which had been ranked as "unnacceptable".

However, there are a few errors with this approach: firstly this is an "unnacceptable" dataset – not a "cyberbullying" dataset.

Secondly, since this data is crowdsourced, the quality is questionable. One person who reports fifty messages which are okay will have as large an impact as someone who is being severely bullied and has reported 50 messages in the last year. There may be things in there which we do not feel need censoring or an overly high focus on a specific field which would make us less likely to pick up on other messages. We would therefore be advised not use this dataset without data cleaning.

The solution to both of these problems is to *use this dataset* to build a Naïve Bayes predictor to filter messages and send us ones which have a high (or nontrivial) liklihood of being "unnacceptable" and then we can manually classify these as "bullying" or not to ensure that the data we are training on is actually very good quality. In this way we can quickly and efficiently source large amounts of data with gold standard tags.

Since we know that the overwhelming majority of messages sent on social media are acceptable; manually trawling through messages to discover which are and

are not acceptable would be a very long and tedious task – so filtering out the obviously acceptable messages using a Naïve Bayes classifier has the potential to speed up data-gathering by many orders of magnitude.

Now that we have a dataset which we know is not okay we can automatically create a lexicon which assigns each word with either a "OK" or a "cyberbullying" sentiment. We should also take care to remove any proper nouns (names) from the lexicon – adding people's names to a lexicon would be an example of overtraining – on test data it may give a higher accuracy however on real-world data it would not.

(iii) How can you use lexicon-based information to make your classifier more robust towards language change?

There are a few ways one could make the lexicon reslient to language change – they all have their flaws.

- Whenever you classify something as "cyberbullying", keep a track of any the words in the data which are not associated with cyberbullying. We should record these and then give the the most frequent ones to a human to see whether they are words related to cyberbullying or not.
- We could update your lexicon based on an online lexicon – this means that the lexicon will never be out of date since *someone else* is already updating it. However, it is also not necessarily tailored to cyberbullying on our specific platform. This would mean that although the quality of prediction is unlikely to decrease, the initial quality would be low to begin with.