Predicting the probability of the breast cancer in patients

Team members

Brendan Lim (A0216513N) **Wong Cheuk Wah** (A0280543H) **Laura Do** (A0280548X)

Outline of this report

- The curse of dimensionality
- Business problem: Prediction breast cancer in patients
- Logistic regression
- K nearest neighbors
- Our conclusion

Curse of dimensionality

What is the curse of dimensionality and how we can show it with our dataset

The curse of dimensionality…

…refers to working with high-dimensional data which is due to large number of features

…occurs during analyzing the data to identify patterns and training the model

…is the model's decreasing performance with more features leaving all else constant as it becomes harder to generalize

We are aiming to show the curse in our dataset by…

… reducing and adding features into our models and observing model performances in each steps

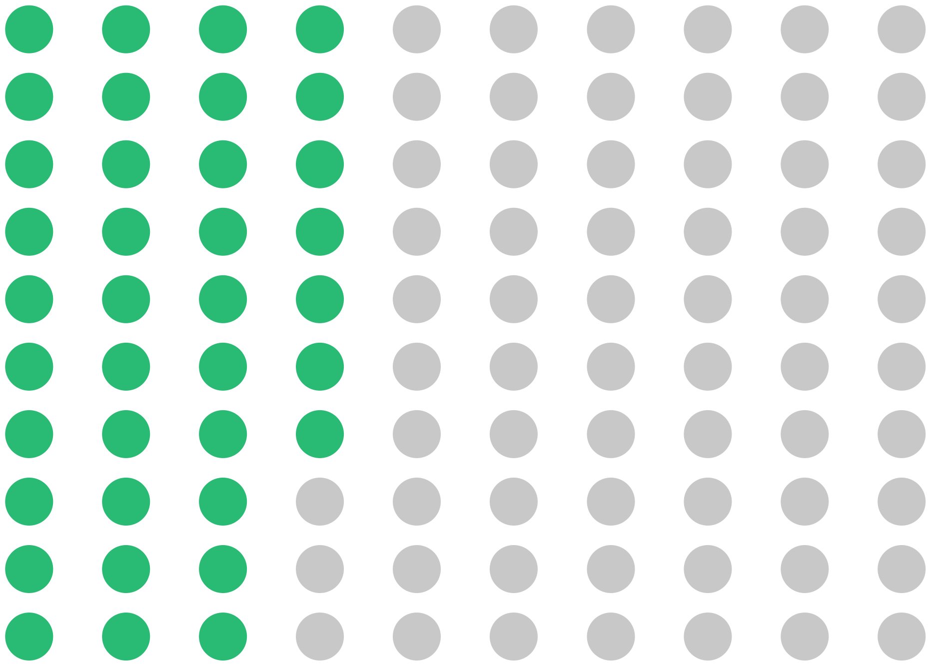
… showing that the model's ability to accurately predict the target is better with fewer than more features

Predicting breast cancer

Based on the breast cancer data, we try to predict the probability for breast cancer…

…using two models

Breast Cancer data set



37% Breast Cancer patients

No Breast Cancer patients 63%

569 data points

30 features

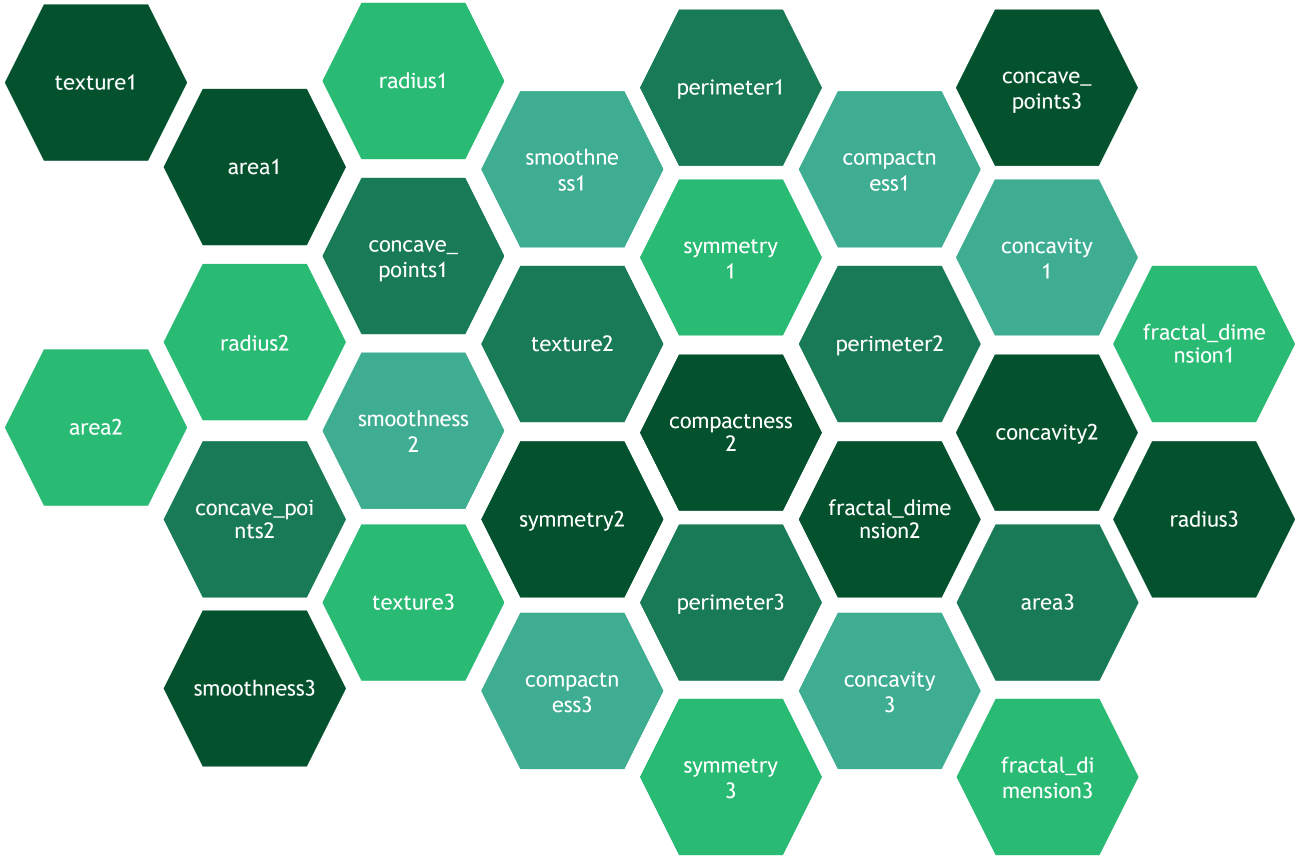
1 target

1. Logistic Regression
2. K nearest neighbors

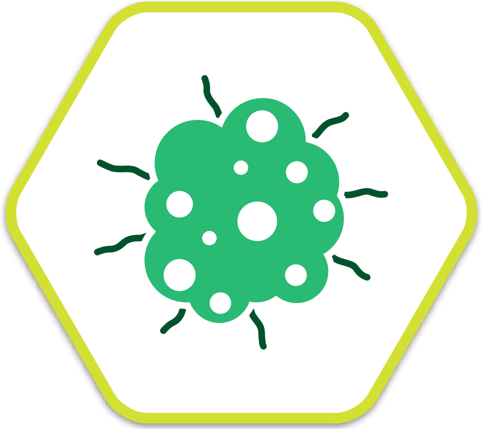
Link to dataset: https://archive.ics.uci.edu/dataset/17/breast+cancer+wisconsin+diagnostic

We have 30 features available to build our classification prediction models

Our dataset provides us with **30 features…**



… which we can **progressively add** to our models…



…to **predict breast cancer** in patients

In the following, we will explore by progressively increasing the dimensionality

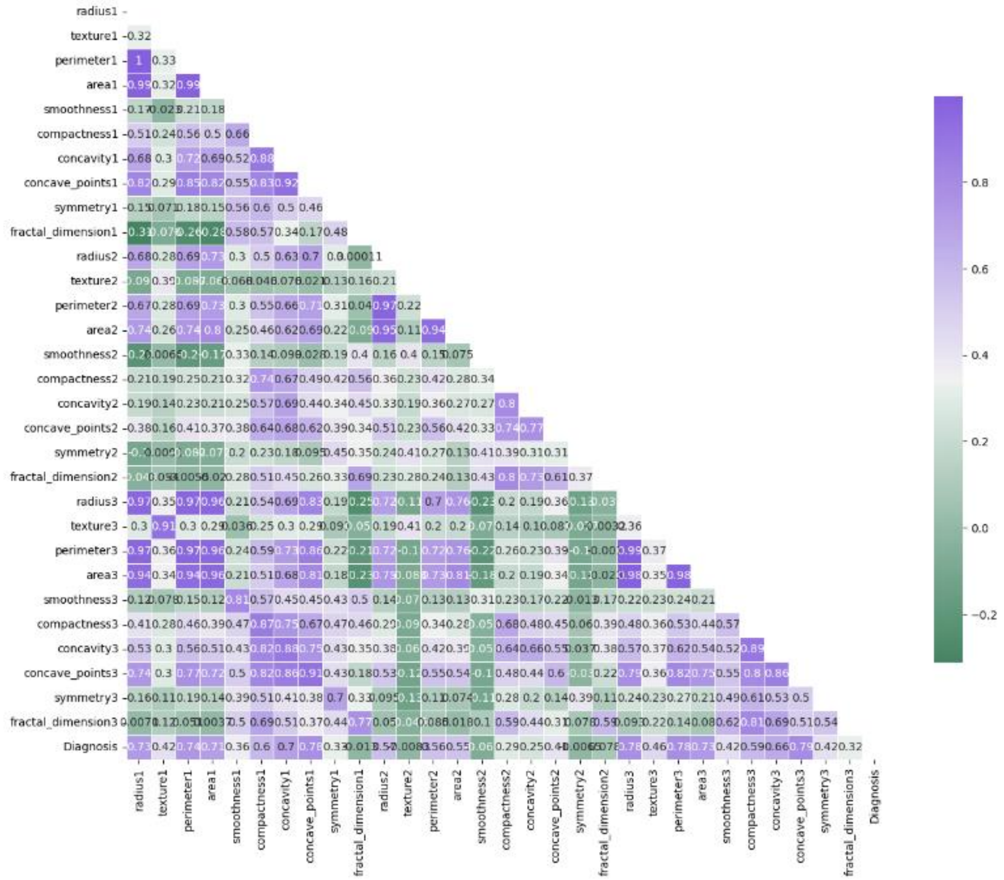
For both the logistic regression and KNN models

1. We are sorting our features in a meaningful order,
2. Then progressively adding the next best feature into our model.
3. As we have 30 features, we train and fit our models 30 times, each time with one more feature.
4. Progressively adding one features after another allows us to see the models' performance while increasing the number of features (= dimensions).

Please see on the following slide for the order

To order the features in a meaningful order to later progressively add them to the models, we look at their correlation to the target 'Diagnosis'

Correlation between 30 features and target 'Diagnosis'



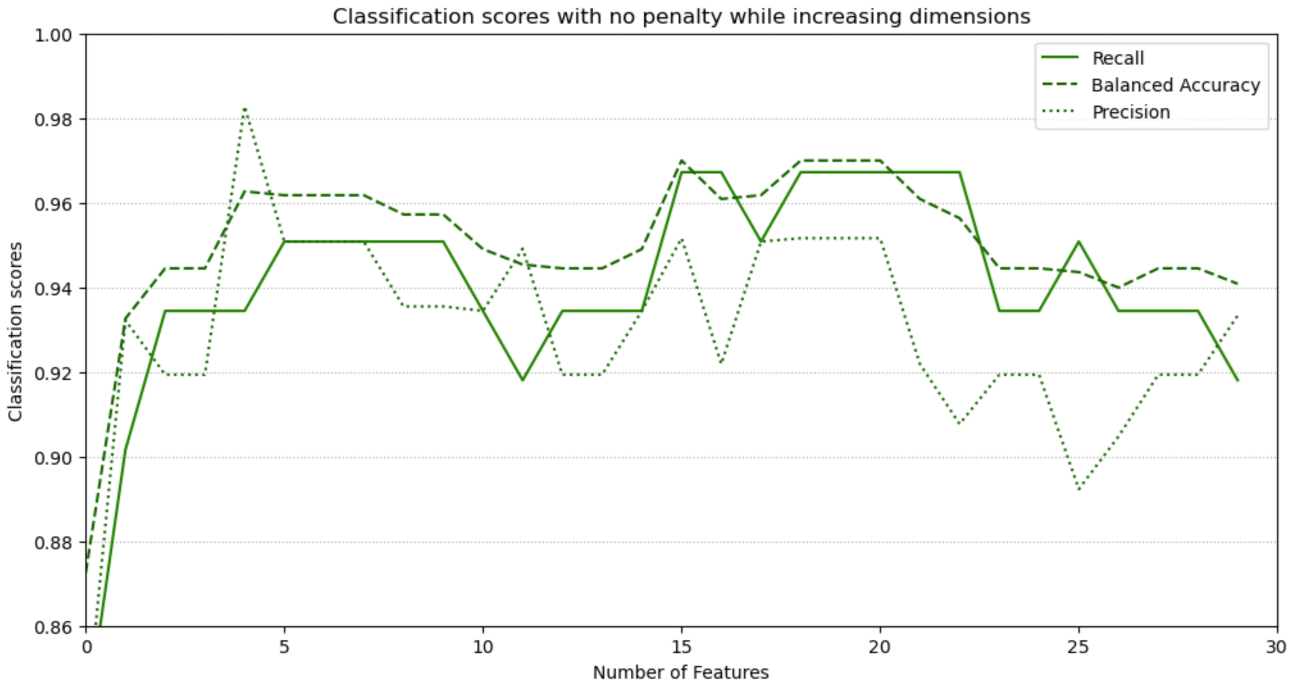
Features ordered according to their correlation

Diagnosis	1.000000
concave_points3	0.793566
perimeter3	0.782914
concave_points1	0.776614
radius3	0.776454
perimeter1	0.742636
area3	0.733825
radius1	0.730029
area1	0.708984
concavity1	0.696360
concavity3	0.659610
compactness1	0.596534
compactness3	0.590998
radius2	0.567134
perimeter2	0.556141
area2	0.548236
texture3	0.456903
smoothness3	0.421465
symmetry3	0.416294
texture1	0.415185
concave_points2	0.408042
smoothness1	0.358560
symmetry1	0.330499
fractal_dimension3	0.323872
compactness2	0.292999
concavity2	0.253730
fractal_dimension2	0.077972
smoothness2	0.067016
fractal_dimension1	0.012838
texture2	0.008303
symmetry2	0.006522
Name: Diagnosis, dtype: float64	

We will pass the features one by one according to this order into our logistic regression and KNN models and observe their performance in prediction breast cancer in patients

Logistic Regression

The baseline logistic regression suffers the curse of dimensionality



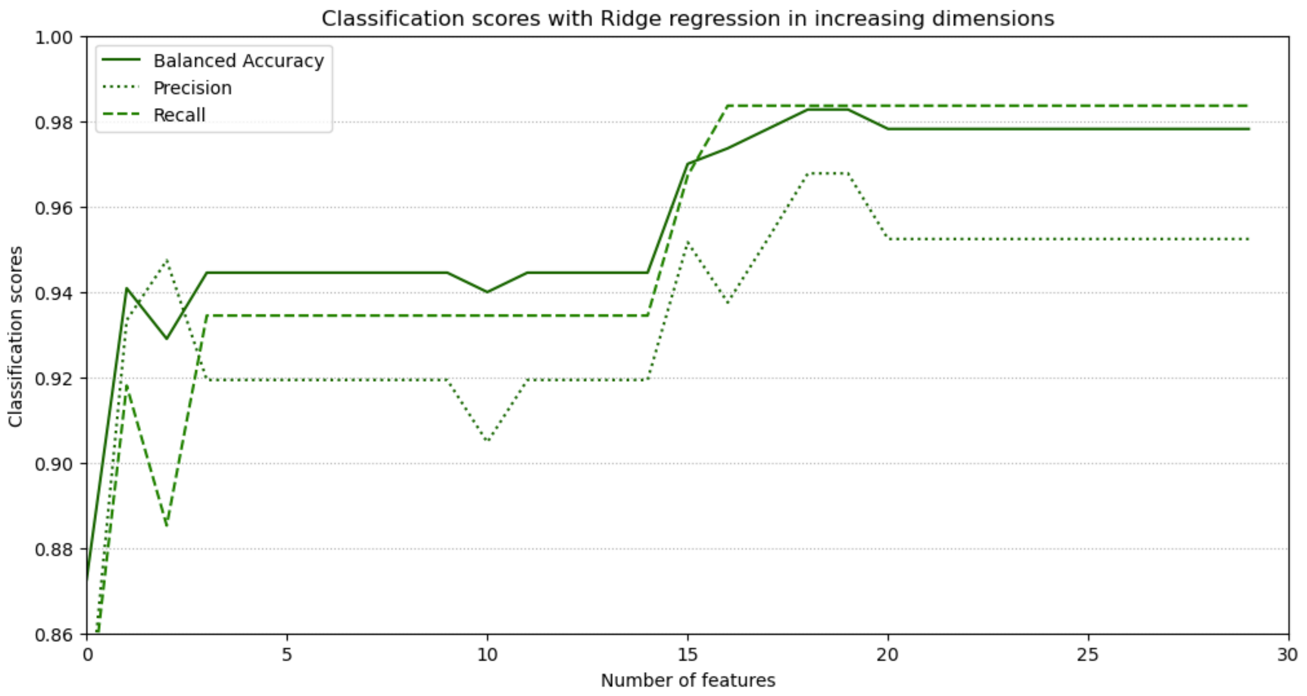
The baseline logistic regression model does **not penalize the coefficients**

While we pass more and more features into the model, the **classification scores**

- improve up until approx. 5 features
- decrease starting from approx. 20 features

Hence, after the 20^{th} dimension we observe the **model suffering from the curse of dimensionality**

Regularizing with Ridge regression does reduce effect of the curse of dimensionality



Introducing Ridge regression into our baseline model **penalizes coefficients**

While we pass more and more features into the model, the **classification scores**

- keep improving until the 20th dimension
- and manages to retain classification scores beyond

Hence, when regularizing with Ridge regression, we do not observe the model suffering from the curse of dimensionality anymore

We are choosing to look at Recall as we think this metric is most important for doctors

Definition of recall

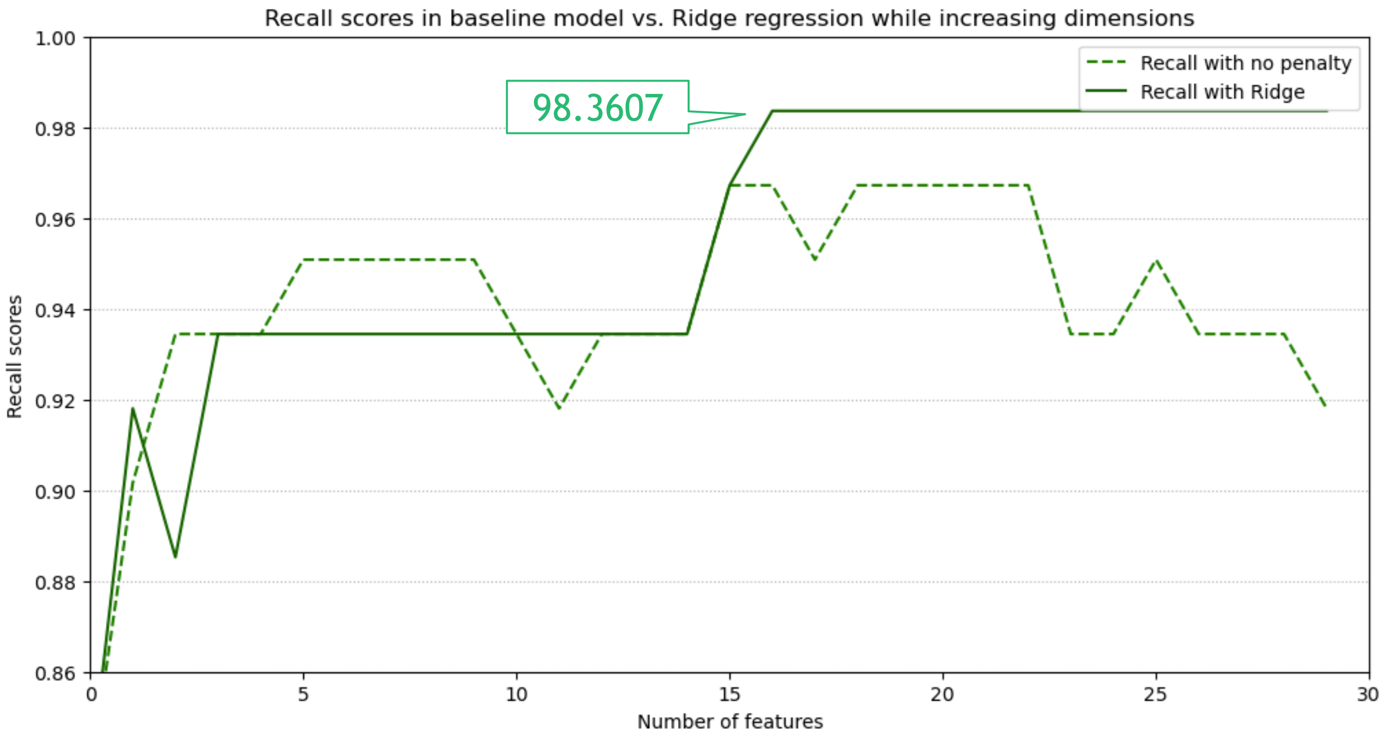
"Out of all the positive examples, how many are predicted as positive?"

$$recall = \frac{tp}{tp + fn}$$

For this business problem,
we choose to primarily compare recall to minimize the number of patients who do have breast cancer, but it goes undetected

Hence, we prefer false positives over false negatives

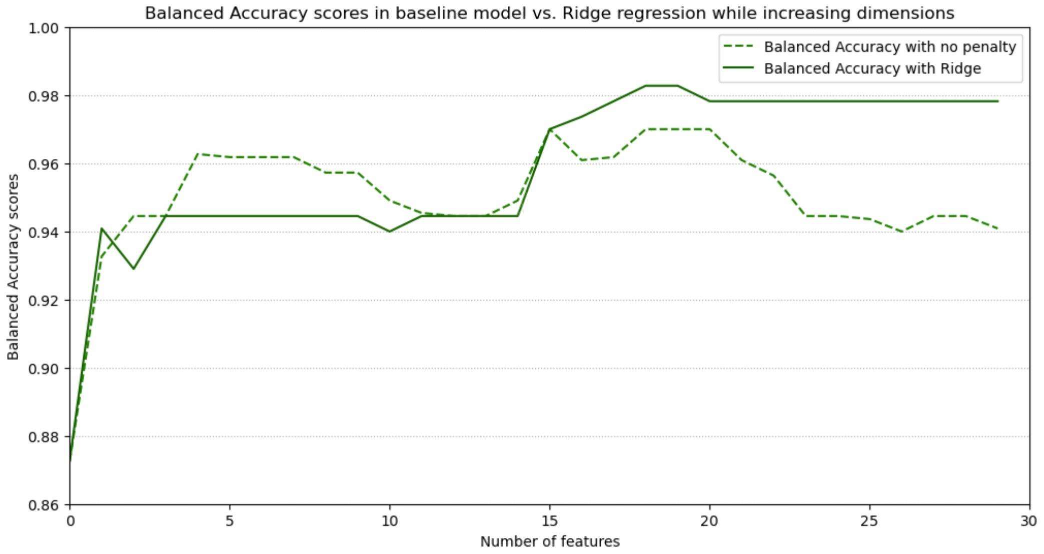
Comparing Recall from both models, we can more clearly see the success in addressing the curse of dimensionality



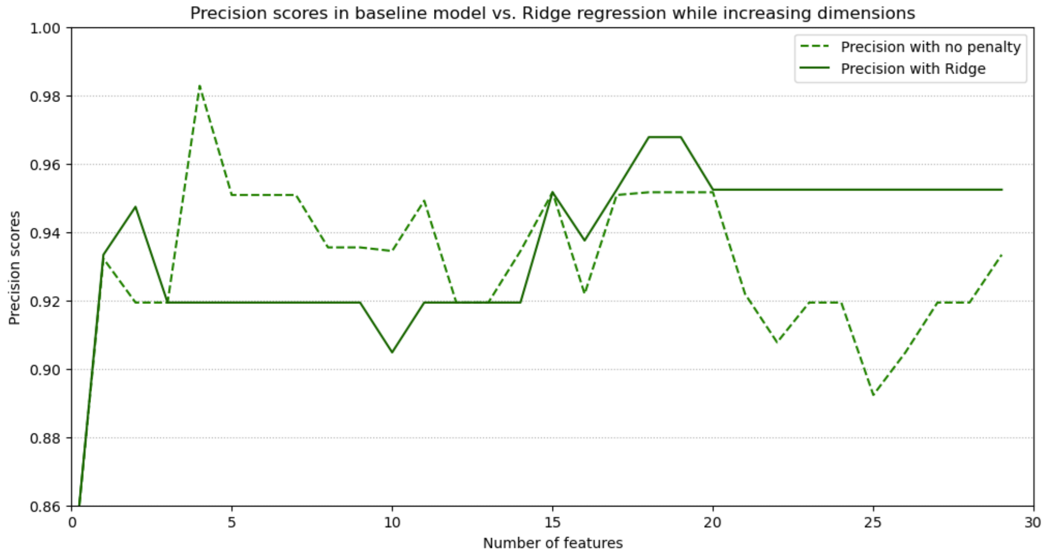
- Looking at Recall, the logistic regression which regularizes with **Ridge does perform better than the logistic regression with no penalty** on coefficients
- Starting from and **beyond the 15th dimension**, regularization does help the model to quite accurately predict whether a patient has breast cancer or not

We yield the same results from looking at Balanced Accuracy and Precision

Observing the curse of dimensionality with Balanced Accuracy starting from the 15th dimension



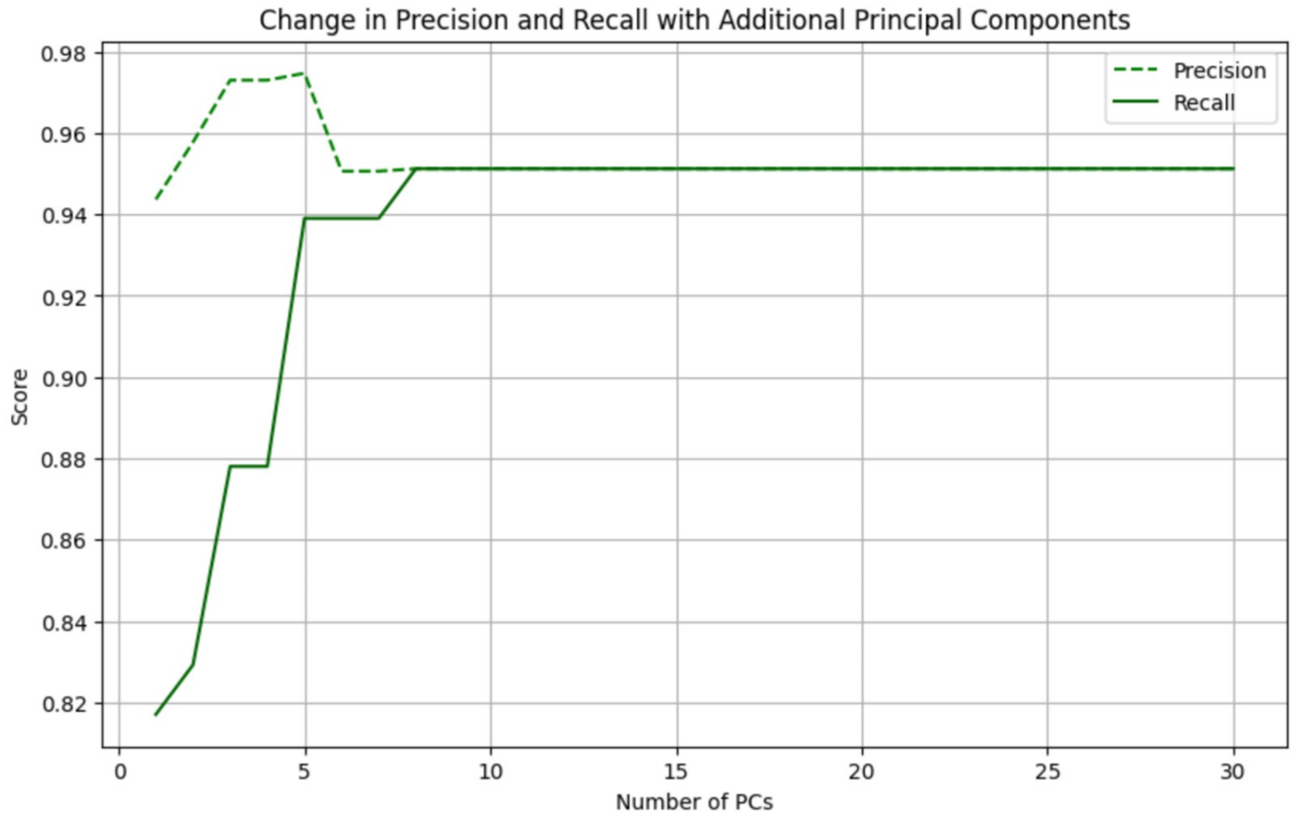
Looking at Precision, the curse of dimensionality starts around the 20th dimension



The PCA method is also successful in addressing the curse of dimensionality

Conducting the PCA method to address the curse of dimensionality,

- PCA contains information from all 30 columns
- we see that Precision and Recall scores stay constant at a high score of approx. 95 after the 8th principal component
- we interpret that it sufficient to extract the first 8 principal components to sufficiently capture the variation of the data



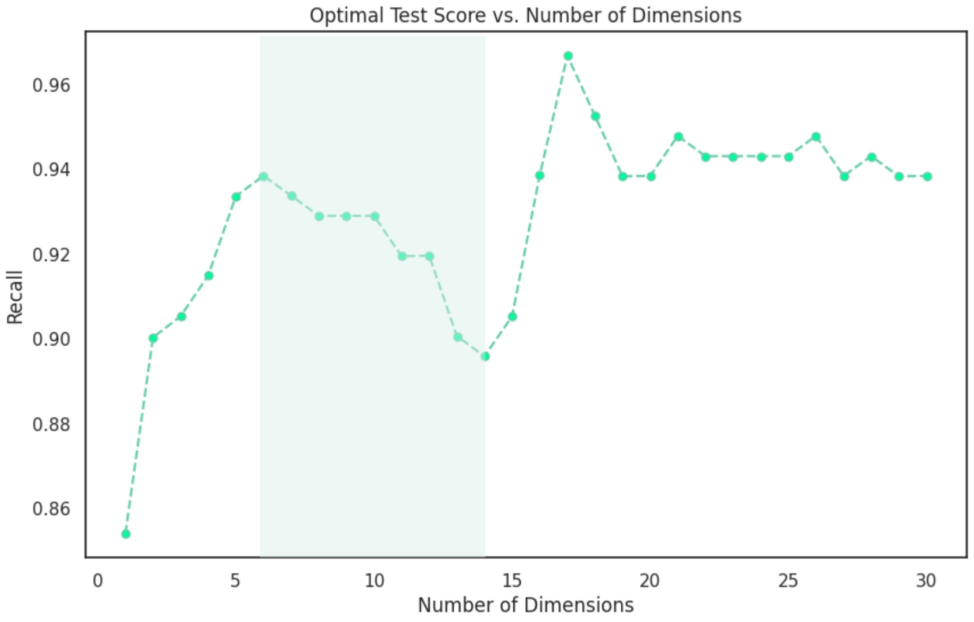
K-nearest neighbors

The baseline KNN model also suffers the curse of dimensionality

Best K values when dimensions are higher

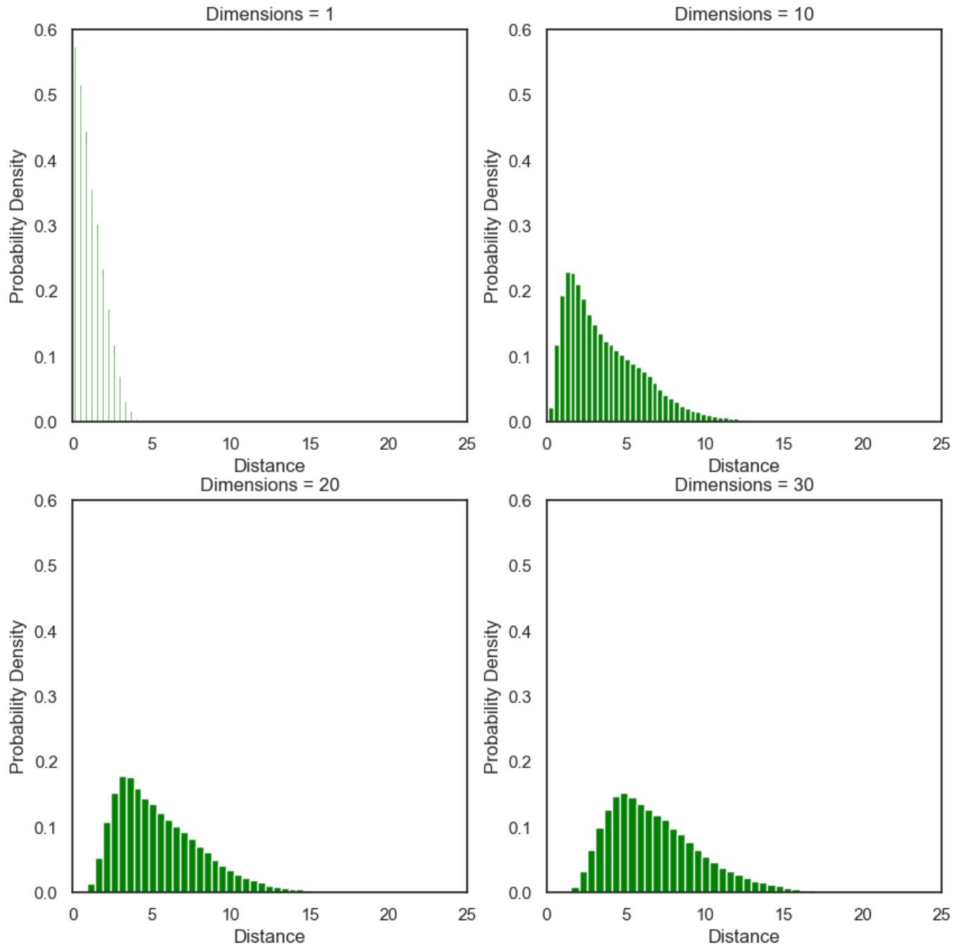
	Number of Dimensions	Best K	Recall Score
0	1	1	0.854042
1	2	1	0.900332
2	3	5	0.905316
3	4	13	0.914950
4	5	3	0.933666
5	6	3	0.938427
6	7	3	0.933776
7	8	3	0.929014
8	9	3	0.929014
9	10	5	0.929014
10	11	3	0.919491
11	12	3	0.919601
12	13	3	0.900664
13	14	3	0.895903
14	15	3	0.905316
15	16	1	0.938538
16	17	1	0.966777
17	18	1	0.952602
18	19	3	0.938317
19	20	1	0.938427
20	21	1	0.947841
21	22	1	0.943079
22	23	1	0.943079
23	24	1	0.943079
24	25	1	0.943079
25	26	1	0.947841
26	27	1	0.938427
27	28	1	0.943079
28	29	3	0.938317
29	30	3	0.938427

Model performance under different dimensions



- Adding dimensions induces noisy and less informative data, leading to a temporary decrease in performance
- Correlation strength may not guarantee all features are equally relevant

In higher dimensions, the data points are located further apart from each other

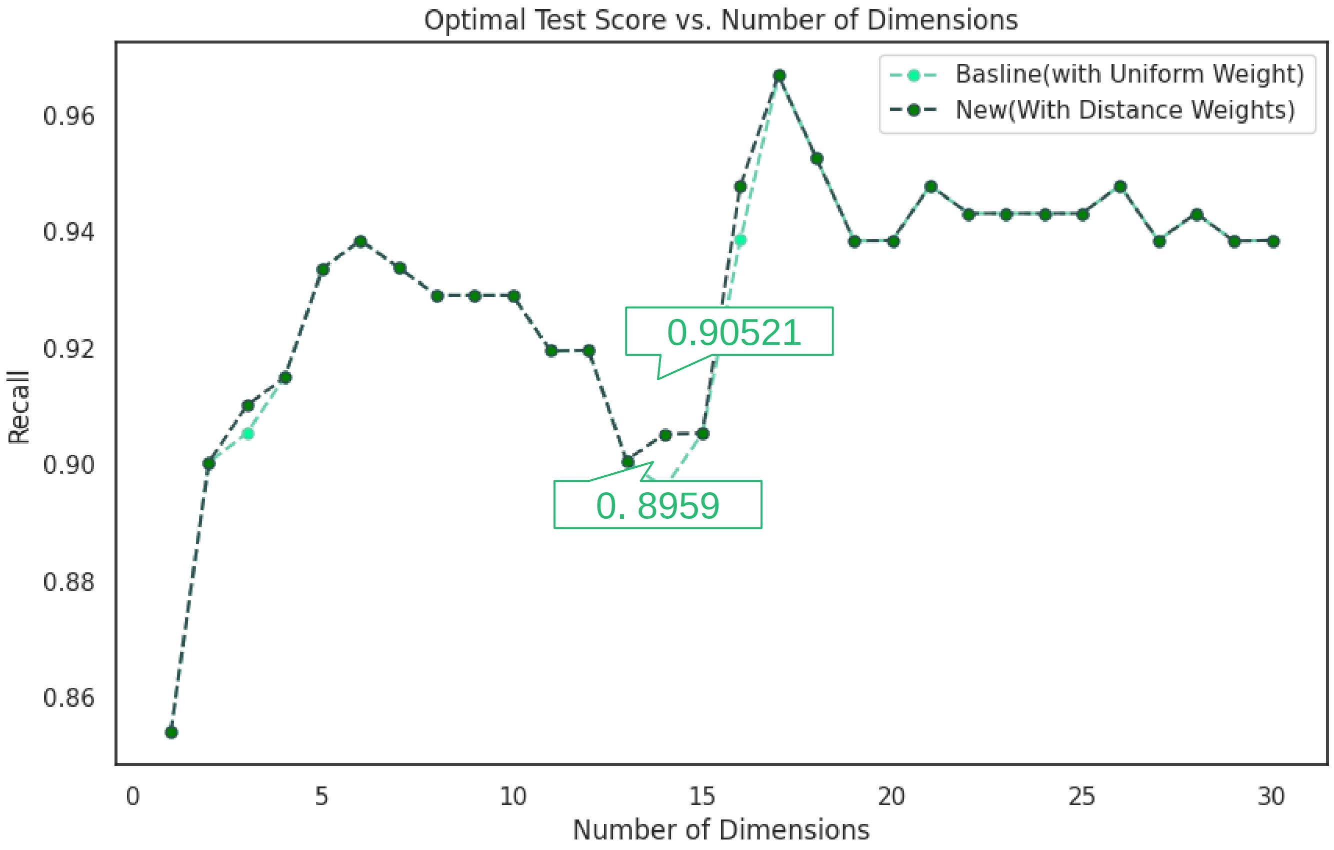


When calculating the **distance between the data points to their neighbors**

- we see that the **distance between them increases** as we move into higher dimensions.
- In one dimension, we see that most distances are very small (approx. 0)
- In 30th dimension, the mean of distances is around 5 and 6

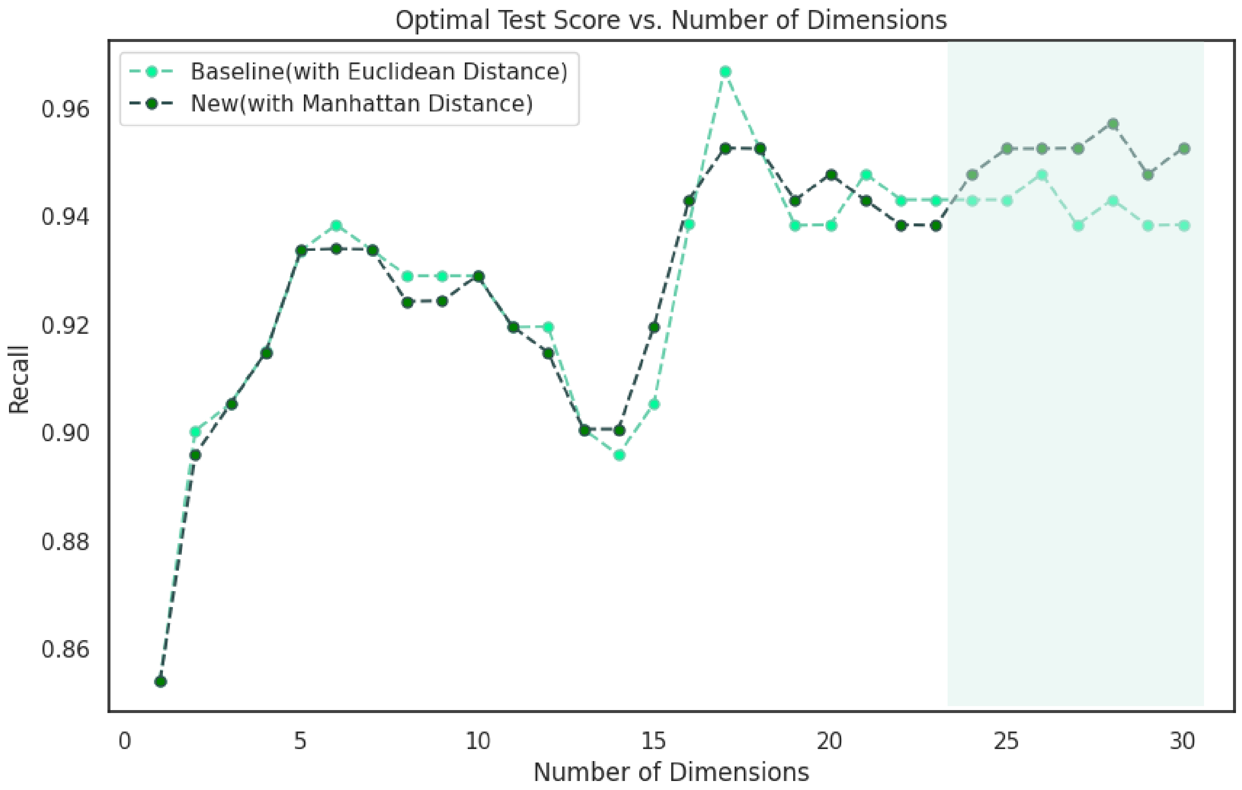
As the distance between data points increases in higher dimensions, it will become **more difficult for the KNN model to find the nearest neighbor**

To address the curse of dimensionality, we change the weight function from uniform to distance which slightly improves Recall

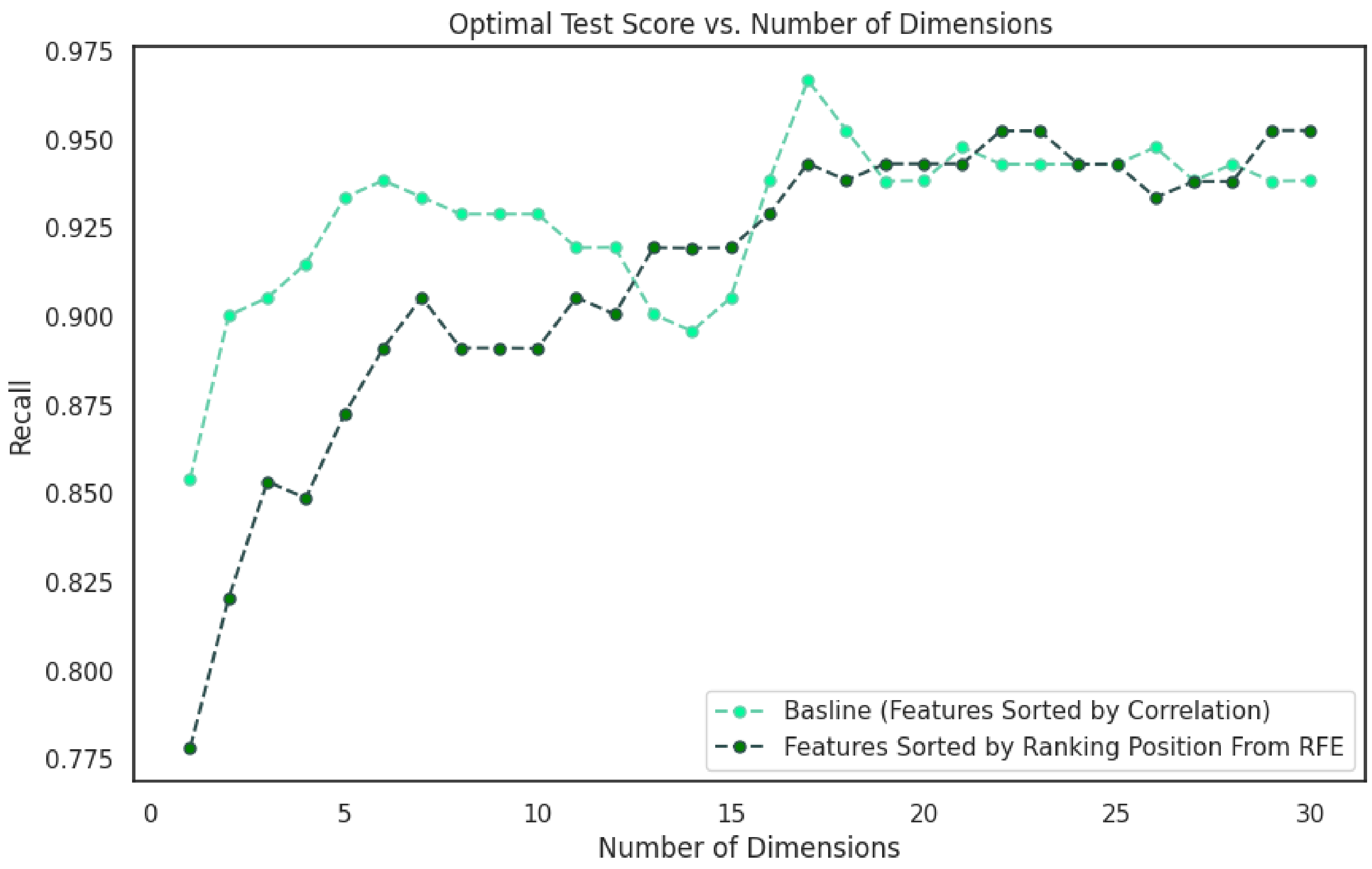


- At first, all points in each neighborhood were weighted equally as the weights are uniform
- With the distance weight function, closer neighbors of a data point will have a greater influence than neighbors further away, hence improving the prediction performance
- This can solely enhance few of the test scores

Manhattan distance yields higher Recall than Euclidean distance when the dimension increases



Sorting features by ranking with Recursive Feature Elimination (REF) effectively mitigates the curse of dimensionality



- All the features are ranked with recursive feature elimination using Logistic Regression as the estimator
- This method identifies the most important features effectively, which enables the recall score to keep increasing even in higher dimensions
- On the other hand, ordering features by their correlation strength to the target introduces noise probably because only linearity is assumed

Conclusion

To predict breast cancer, we conclude that it is better to use logistic regression rather than KNN because …

Comparing logistic regression with KNN

1. **logistic regression performs better** for most N features

2. logistic regression is more flexible as **regularization is an option**, which cannot be conducted for KNN

When feature selection is possible

- we would encourage the **selection of the first N parameters** before recall starts to fall

- LASSO regularization was thus not utilized, as **we acknowledge that in some scenarios:**
  - dimensional reduction may not be a possibility
  - feature selection may have already been done
  - certain features may have to be included due to domain specific knowledge