# Principal Component Analysis

Tymo van Rijn

March 2026

## 1 Introduction

### 1.1 What is PCA?

Principal Component Analyis (PCA) is a method based on linear algebra that allows us to reduce the dimensionality of a (high dimensional) vector and create new features called *principal components*. **PCA** does this based on the variance across the vectors in a vector space to see which of the dimensions are the most "important". **PCA** selects the top components with the highest eigenvalues and projects the data onto them to simplify the dataset. It prioritizes the directions where the data varies the most because more variation means more useful information.

### 1.2 How does it work?

**1. Standardize the data**
If not done already, it is important to standardize the data. In this scenario, this means there should be a mean of 0 and a standard deviation of 1

$$Z = \frac{X - \mu}{\sigma}$$

where:
$\mu$ is the mean of independent features $\mu = \{\mu_1, \mu_2, \ldots, \mu_m\}$
$\sigma$ is the standard deviation of indepentend features $\sigma = \{\sigma_1, \sigma_2, \ldots, \sigma_m\}$

**2. Calculate Covariance Matrix**
Next PCA calculates the covariance matrix, to see which features relate to each other (whether they increase or decrease together). The covariance between two features $x_1$ and $x_2$ is

$$cov(x1, x2) = \frac{\sum_{i=1}^{n}(x1_i - \bar{x1})(x2_i - \bar{x2})}{n-1}$$

where:
$\bar{x1}$ and $\bar{x2}$ are the mean values of features $x_1$ and $x_2$
$n$ is the number of data points

The value of covariance can be positive, negative or zeros

**3. Find the Principal Components**
PCA identifies new axes where the data spreads out te most

- **1st Principal Component (PC1):** The direction of maximum variance (most spread)
- **2nd Principal Component (PC2):** The next best direction, *perpendicular / orthogonal to PCA1* and so on

These directions come from the eigenvectors of the covariance matrix and their importance is measures by eigenvalues. For a square matrix $A$ an eigenvector $X$ (a non-zero vector) and its corresponding eigenvalue $\lambda$ satisfy:

$$AX = \lambda X$$

This means:

- When $A$ acts on $X$ it only stretches or shrinks $X$ by the scalar $\lambda$.
- The direction of $X$ remains unchanged hence eigenvectors define "stable directions" of $A$

Eigenvalues help rank these directions by importance.

**4. Pick the Top Directions & Transform Data**
After calculating the eigenvalues and eigenvectors PCA ranks them by the amount of information they capture. We then select the *top k* components that capture most of the variance like 95% (for example). And then transform the original dataset by projecting it onto these top components.

In theory this should mean that we reduced the number of features (dimensions) while keeping the important patterns in the data.

### 1.3 PCA motivation in our case

One of the advantages of PCA is **noise reduction**, which is something we thing occurs inside of the **FA** embeddings. By applying PCA, we can pick a smaller amount of features that could still resemble our FA exam in a proper way. This could lead to better results in the downstream classification task using a GRU because there is less noise that is being thrown in the pipeline.

## 2 Methodology

I am planning on just throwing the entire examination in the GRU (meaning all of the frames (embeddings) of every exams will be used), no picking of certain bins/frames, to make it as easy and baseline as possible.

First for the baselines results, I will train the GRU with the original (higher-dimensional) embeddings. And obtain results based on **ROC-AUC** and **AP**. These results are $R_B$

I am using the *training data* of the *Angioreport* dataset to perform PCA on.

After I performed the PCA, I will make a charge plot to visualize how many principal components achieve what degree of aggregated variation. Based on these plots, I will make a decision on how many PC's ($n_{PC}$) I want to include for a specific model (this may vary across models). As of right now, I have read that 95% variance coverage is good-practice

When the $n_{PC}$ is concluded, I will prepare the pipeline to project all of the embeddings onto these PC's for the certain model. When that is done these lowered-dimensioned embeddings will be fed to the GRU as a sequence (per examination) and have the downstream classification task for the HyperF_Type set up. These will also result in new metric scores which are also based on **ROC-AUC** and **AP**. These results are $R_P$.

Then I will compare the $R_B$ with the $R_P$ and see if there are noticable differences between those two results, and based on these results we will know how to move forward from here on.

Table 1: Hyperparameters for the GRU classifier.

| **Setting** | **Value** |
|---|---|
| GRU layers | 1 |
| Hidden size | 128 |
| Readout | Last hidden state $\rightarrow$ Linear |
| Optimizer | Adam |
| Learning rate | $10^{-4}$ |
| Batch size | 32 |
| Epochs | 60 |
| Loss function | Cross-entropy |
| Input features | Raw embeddings / PCA |
| Random seeds | 0, 1, 2, 42 |

# 3 Hypotheses

*Since I only just learned the importance of having these, I will create them here as well*

## 3.1 $H_0$ Null Hypothesis

When applying PCA on the embeddings for the angioreport dataset to reduce the dimensionality, the overall performance of the GRU will not benefit and decline in **ROC-AUC** and **AP** results or stay the same.

## 3.2 $H_1$ Performances increases

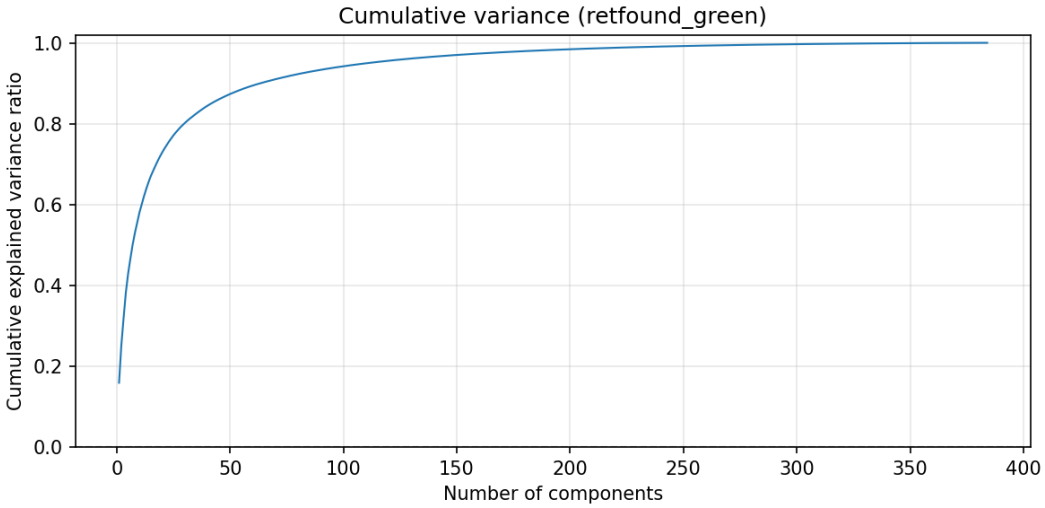
When applying PCA on the embeddings for the angioreport dataset to reduce the dimensionality, the overall performance of the GRU will benefit and reduce noise, resulting in higher **ROC-AUC** and **AP** scores.

## 3.3 $H_2$ Higher dimensional embeddings benefit more

The different models consist of different dimensionalities $384D$ and $1024D$. If the PCA is applied on both of these embedding sizes, dimensionality reduction as a result of PCA, will result in a higher increase in **ROC-AUC** and **AP** scores for $1024D$ embeddings than for $384D$ embeddings.

# 4 Results

## 4.1 RETFound Green

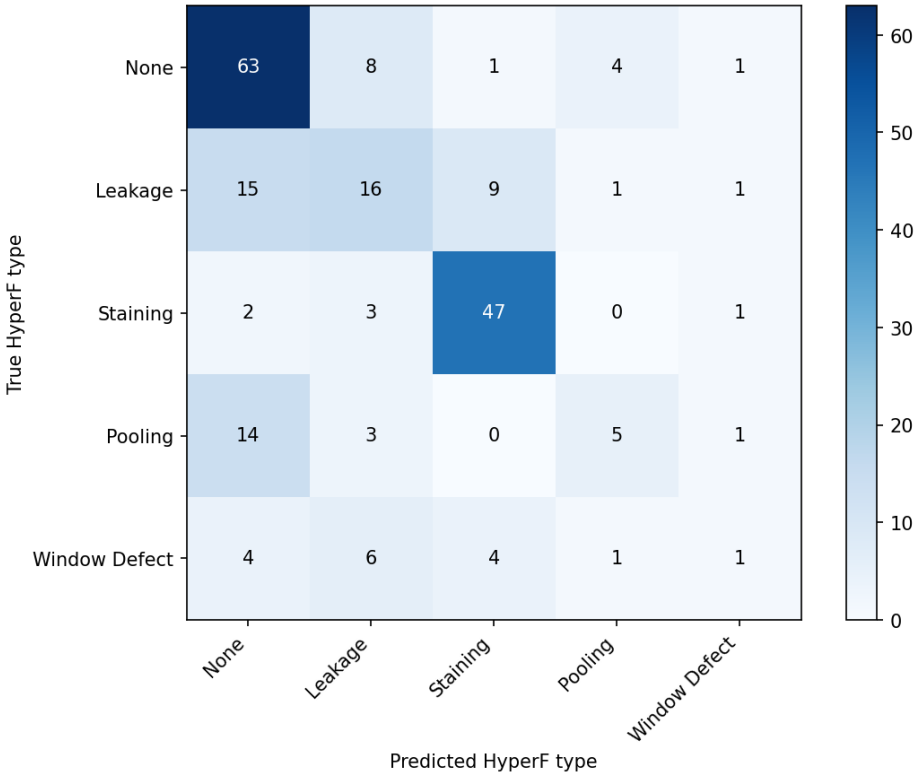



As shown in this charge plot, it is visible that the RETFound-Green PCA gains quite some variance at a "low" level of embeddings. To reach the state of 95% variance coverage, the $N_{PC} = 111$. In order to reach 99% variance coverage, the $N_{PC} = 235$.

| Variance Explained | $N_{PC}$ | macro AUC | macro AP |
|---|---|---|---|
| 0.95 | 111 | 0.78±0.0072 | 0.49±0.0164 |
| 0.99 | 235 | 0.78±0.0064 | 0.5±0.0129 |

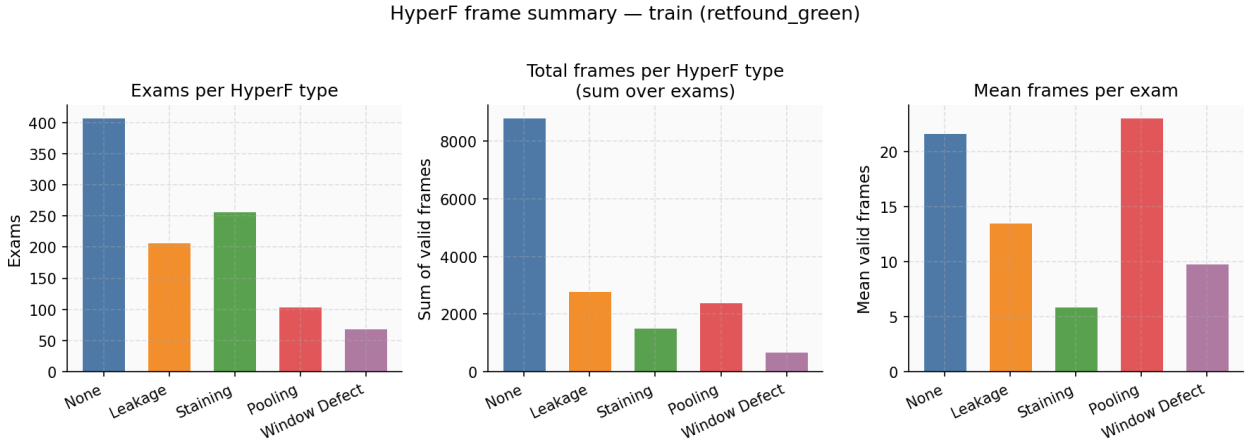
While the $R_B$ achieved an **macro AUC** of **0.8025**±0.0074 and **macro AP** of **0.5108** ±0.0185. So in this case, it doesn't seem like the usage of PC's helped the downstream classification task. Looking at the confusion matrix, it is clear that the underrepresented classes don't get great scores.

Figure 1: Confusion matrix RETFound Green $N_{PC} = 111$

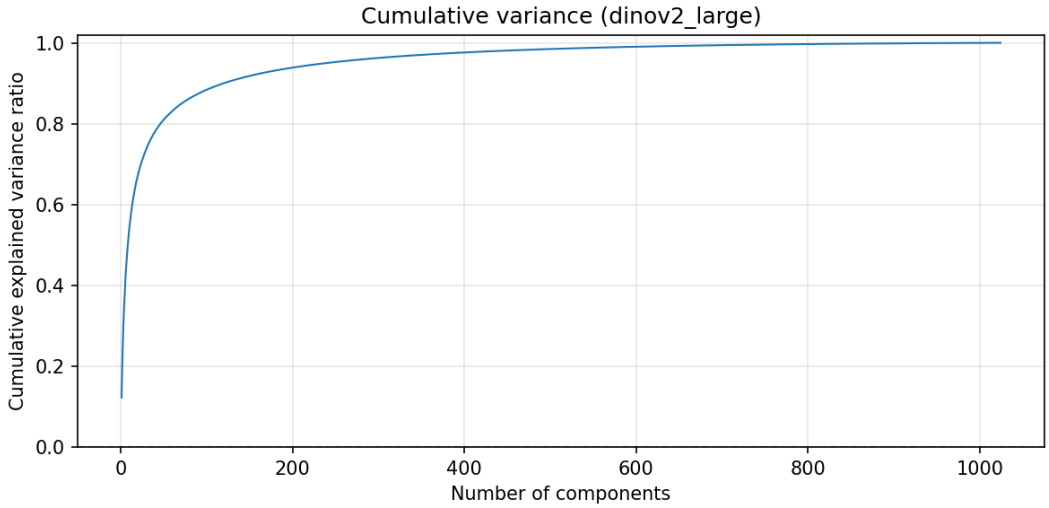



But this has more to do with the frame distribution per class, as shown in *Figure 2*, there is a huge imbalance, not only in exams, but even more so in the total amount frames per HyperF class. So when making use of the entire examination, and therefore using all of the existing frames, the imbalance only increases; None gets the overhand.

Figure 2: HyperF_Type frame summary




## 4.2 Dinov2 Large



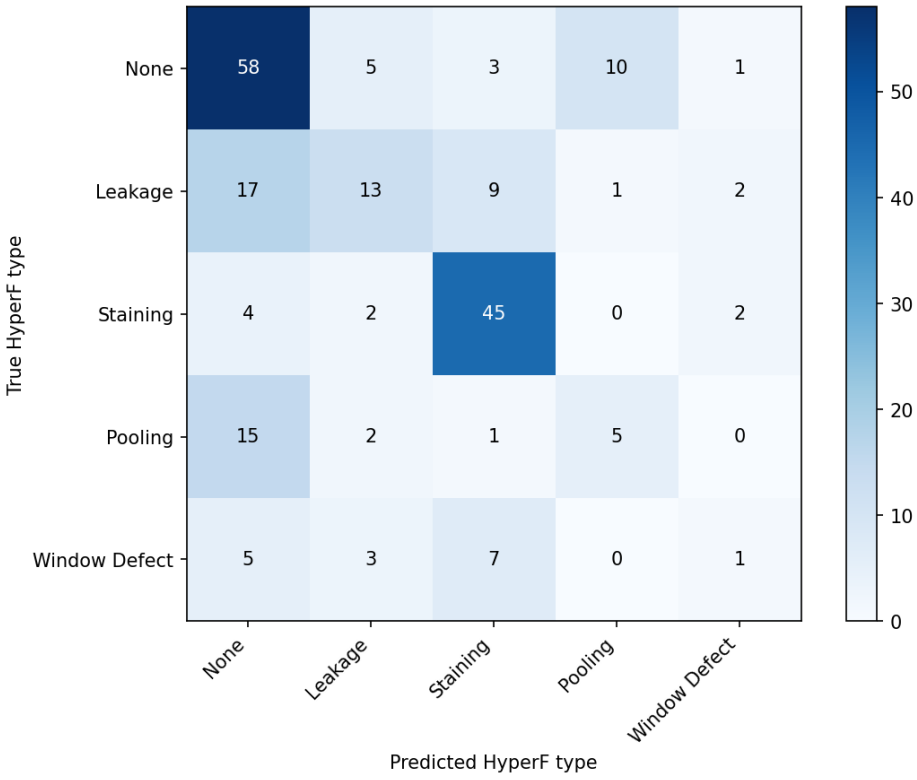

Dinov2 Large shows a similar charge plot as the RETFound-Green's one. In order to get to 95% variance covered for the Dinov2 Large, $N_{PC} = 240$. To get to 99& variance coverage, $N_{PC} = 594$

| Variance Explained | $N_{PC}$ | macro AUC | macro AP |
|---|---|---|---|
| 0.95 | 240 | 0.763 ±0.008 | 0.459 ±0.012 |
| 0.99 | 594 | 0.756 ±0.008 | 0.452 ±0.004 |

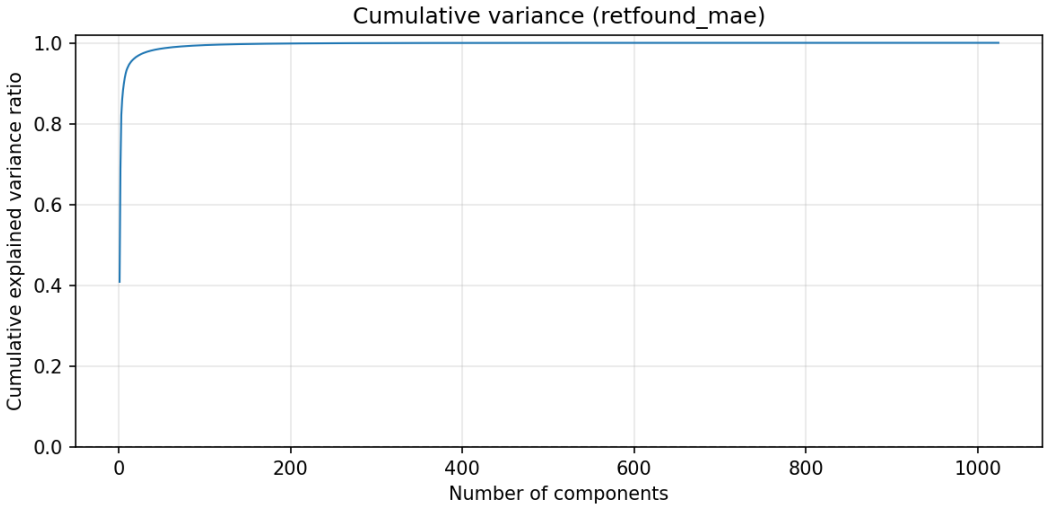
The $R_B$ achieved an **macro AUC** of **0.7821**±0.0077, and **macro AP** of **0.468**±0.014. So in this case, it doesn't seem like the usage of PC's helped

the downstream classification task. Looking at the confusion matrix, it is clear again that the underrepresented classes don't get great scores.

Figure 3: Confusion matrix Dinov2 Large $N_{PC} = 594$



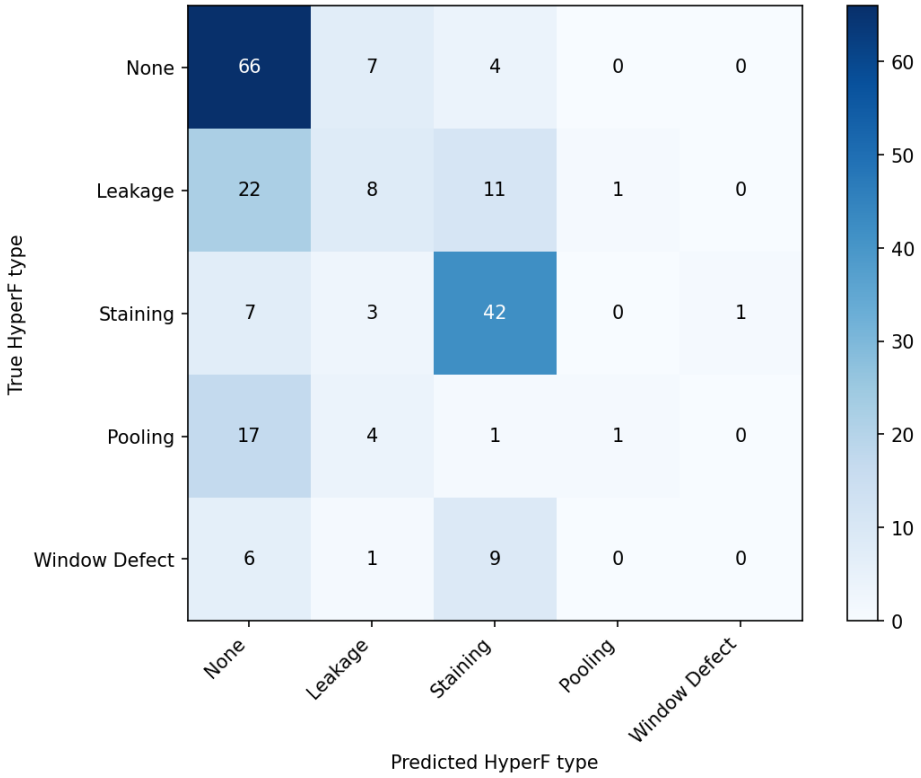
### 4.3 RETFound MAE




RETFound MAE shows a steeper plot than the first two. In order to get to 95% variance covered for the RETFound MAE, $N_{PC} = 14$. To get to 99& variance coverage, $N_{PC} = 67$. These are substantially lower than the first 2 models.

| Variance Explained | $N_{PC}$ | macro AUC | macro AP |
|---|---|---|---|
| 0.95 | 14 | $0.7274 \pm 0.0014$ | $0.3981 \pm 0.0163$ |
| 0.99 | 67 | $0.7421 \pm 0.0117$ | $0.4170 \pm 0.0111$ |

The $R_B$ achieved a **macro AUC** of **0.7682**$\pm$0.0037 and a **macro AP** of **0.4480**$\pm$0.002. Again, the "noise reduction" did not do much, this could be because it does not yet know what noise is in this specific classification task.




# 5 Reflection

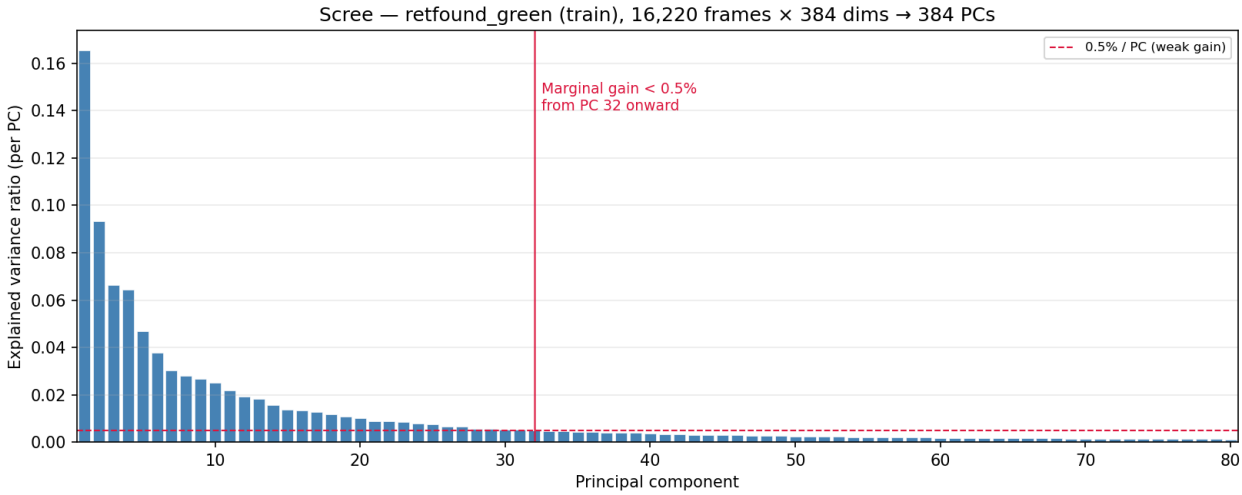
After I did some more deepdiving into PCA, I found out that using a scree plot to determine the amount of PC's you should probably be using, would be a better way than just use 95- and 99% of the variance covered.

The Scree plot allows you to apply the "Elbow method". This method implies that you should look at the rate of change, if the 10th component explains 5% of the variance, and the 11th explains 0.1%, that "elbow" is a natural cutoff point regardless of the total cumulative variance.

This is a mistaeke that could have been prevented if I took a bit more time into finding ways to base my $N_{PC}$ up on. I am not saying what I did right now was wrong, but I do like this other approach more, since it asks you to look at every model specific and make a thought of decision.

So right now, I will do it for RETFound-Green, and see what comes out of it, if it hopeful, I might do the other ones as well (depending on what the feedback is from my company supervisor)

# 6 Extra (scree plot & run)




This is the scree plot for RETFound Green embeddings, I decided that when the gain per PC goes down to $< 0.5\%$, it is not useful anymore. That resulted in stopping the accumulation at PC 32, from this PC onward it did not seem like there was a lot of variance being added, therefore ***maybe*** just containing noise (I don't know).

32 PC's is a relatively low number of $N_{PC}$, compared to the first method for RETFound green, where the respective $N_{PC}s$ were 111 and 235.

| Variance Explained | $N_{PC}$ | macro AUC | macro AP |
|---|---|---|---|
| NaNN | 32 | $0.7802 \pm 0.0084$ | $0.4742 \pm 0.0055$ |

# 7 Conclusion (for now) + Reflection

In this side experiment, I have concluded that using PC's does not improve the downstream performance when using a GRU. This could be the result of the embeddings not being trained on our specific classification task (HyperF_type), or the clinical information resides in those later PC's. It is not new, that with even high amounts of variance covered, the results of a model go down [1].

I have also concluded that writing a "report" like this, without making use of an LLM to polish your own-written text, was quite challenging, and resulted in a somewhat messier result. It goes to show that my usage of AI over the past year(?), really worsened my ability to write simple texts, so I think it is time to take matter into my own hands and start using my own brain more.

## References

[1] Jianwei Zheng and Cyril Rakovski. On the application of principal component analysis to classification problems. *Data Science Journal*, 20(1):26, 2021.