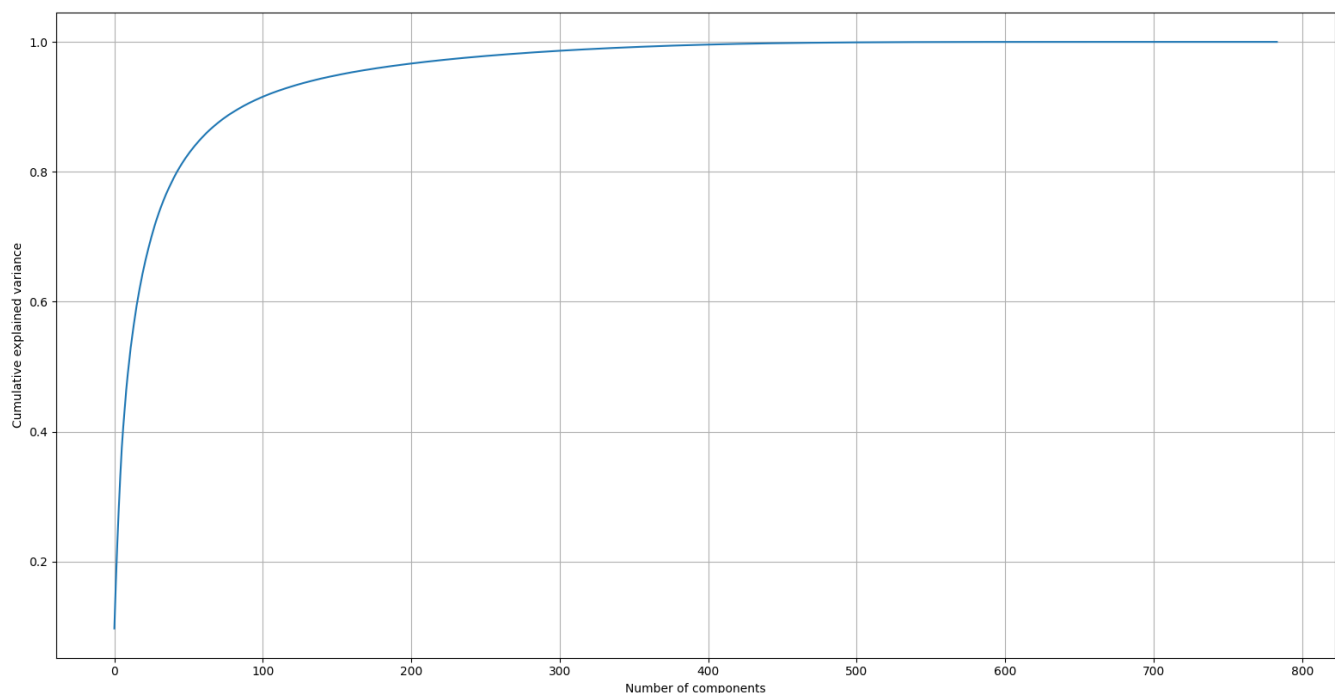


- Задача - добиться улучшения качества работы модели, основанной на персептроне, путем изменения dropout, функции активации и оптимизационного алгоритма
- Руководство воспроизведения работы кода для подбора гипер-параметров (команды выполняются в терминале терминале)
  - **pip install -r requirements.txt** (в новом виртуальном окружении)
  - **python optuna\_pytorch.py**
  - **tensorboard --logdir runs**
  - Проследовать в браузере по адресу <http://localhost:6006>, пользуясь доступным функционалом ознакомиться с визуализацией процесса подбора параметров
- График зависимости совокупно-объясненной дисперсии от главных компонент датасета



- На основании графика принято решение провести подбор параметров, ограничив главные компоненты между 100 и 200, с шагом в 20
- Точностью которой удалось достичь - 0.76
  - Количество скрытых слоев - 1
  - Размерность скрытых слоев - 60
  - Количество эпох - 20
  - Уровень обучения - 0.05
  - Функция активации - glu
  - dropout (доля исключенных нейронов) - 0.0
  - Оптимизационный алгоритм - Adagrad
- Вывод - применение PCA позволяет значительно снизить вычислительную сложность без серьезных потерь в точности и проверять рабочие гипотезы в

краткие сроки на слабых машинах, что сэкономит деньги бизнеса и поможет создать MVP быстро и дешево.