Feature learning using matrix factorization and neural networks

Miami Machine Learning Meetup
January 2018

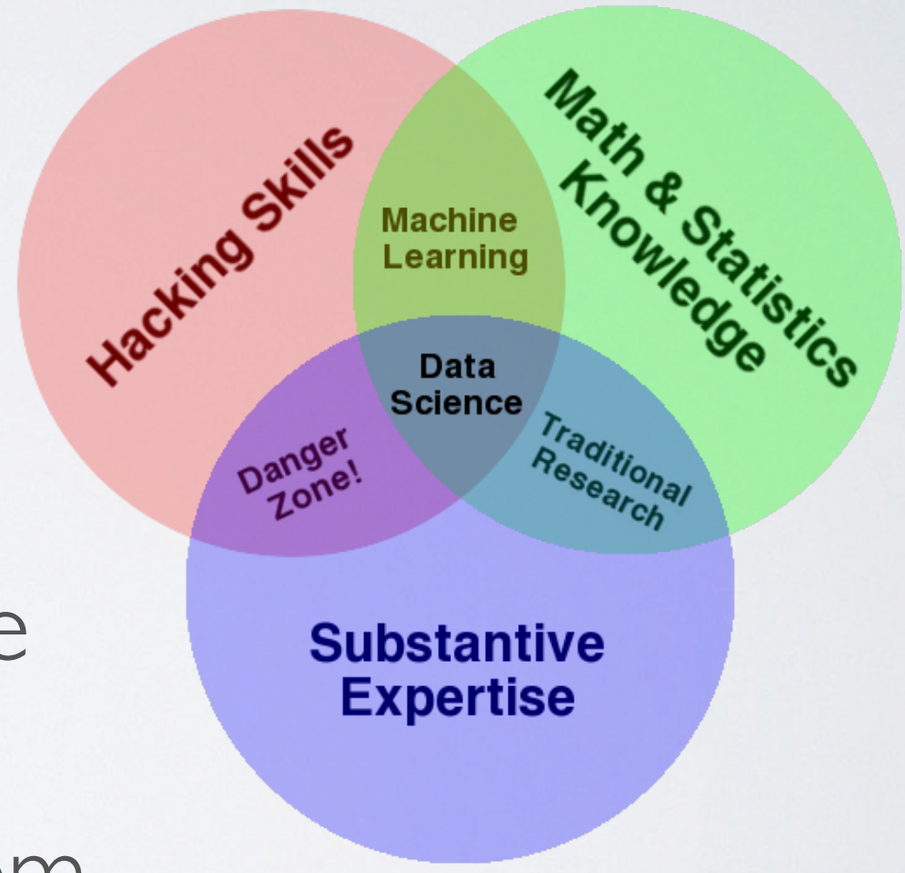
* record

About Me

MODERNIZING MEDICINE

FAU
FLORIDA ATLANTIC UNIVERSITY

- Aaron Richter
- Data Scientist
  - Thrive in the danger zone
- PhD Candidate, Computer Science
- @rikturr rikturr@gmail.com



What we'll cover

- What is feature learning?
- Why feature learning?
- How to do feature learning?
  - Matrix Factorization
  - Neural networks
- Code samples

What is feature learning?

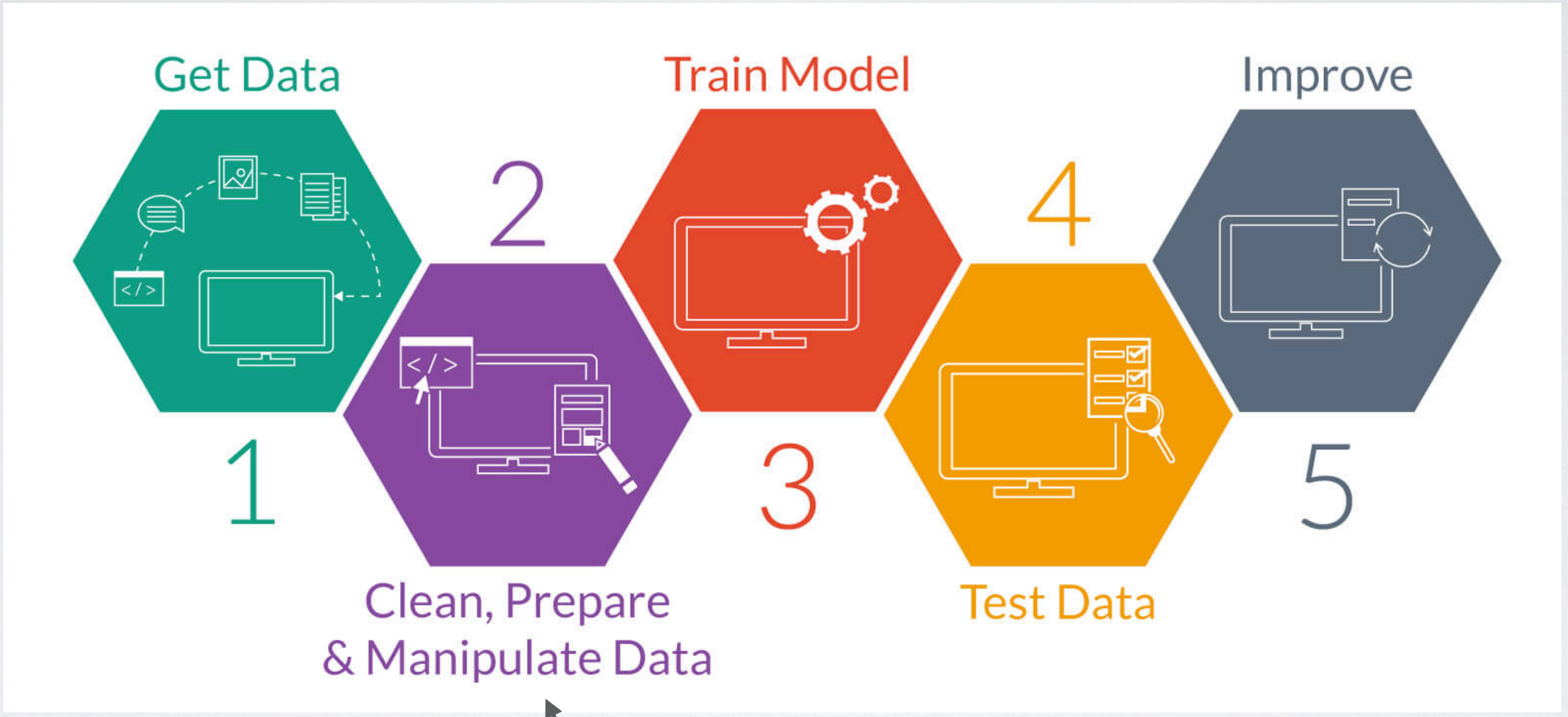
- Automatically extracting features from raw data using machine learning techniques

- Create latent feature space for input to predictive models

- Also known as feature embedding, feature extraction, representation learning

https://en.wikipedia.org/wiki/Feature_learning

Why feature learning?



Feature engineering

https://upxacademy.com/introduction-machine-learning/

Why feature learning?

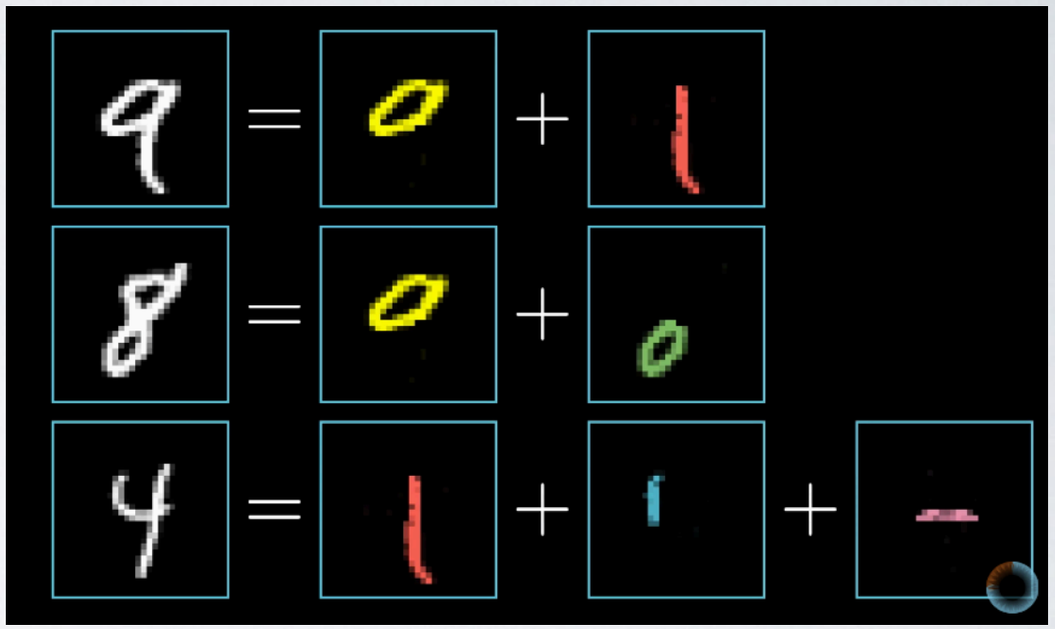
- Feature engineering is hard!
- “Coming up with features is difficult, time-consuming, requires expert knowledge. Applied machine learning is basically feature engineering.” - Andrew Ng

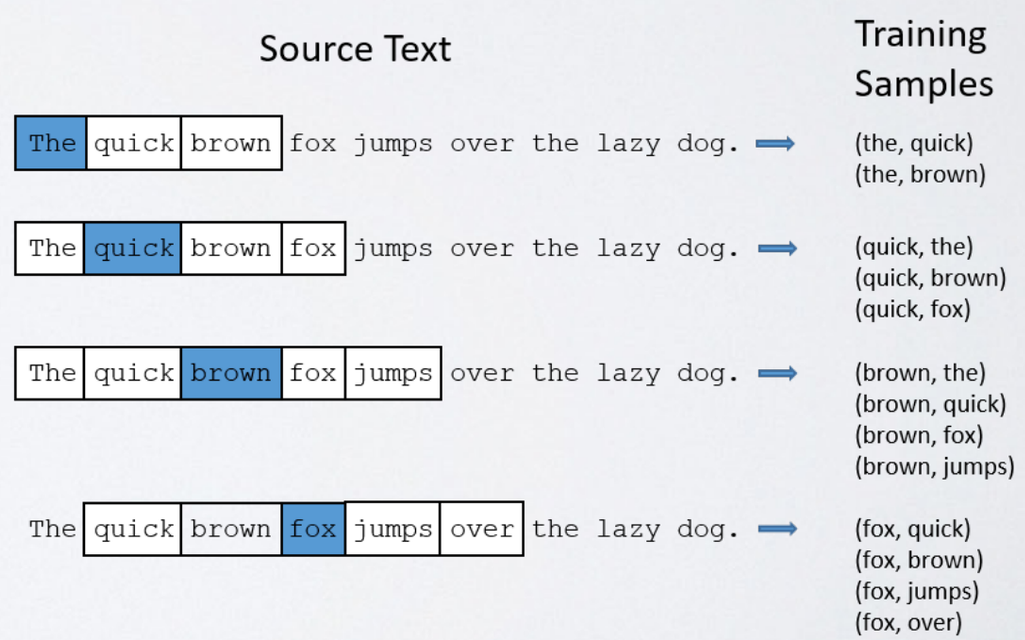
https://en.wikipedia.org/wiki/Feature_engineering

https://forum.stanford.edu/events/2011/2011slides/plenary/2011plenaryNg.pdf

Example applications

- Computer vision
- Text mining
- How do you tell the computer what to look for?



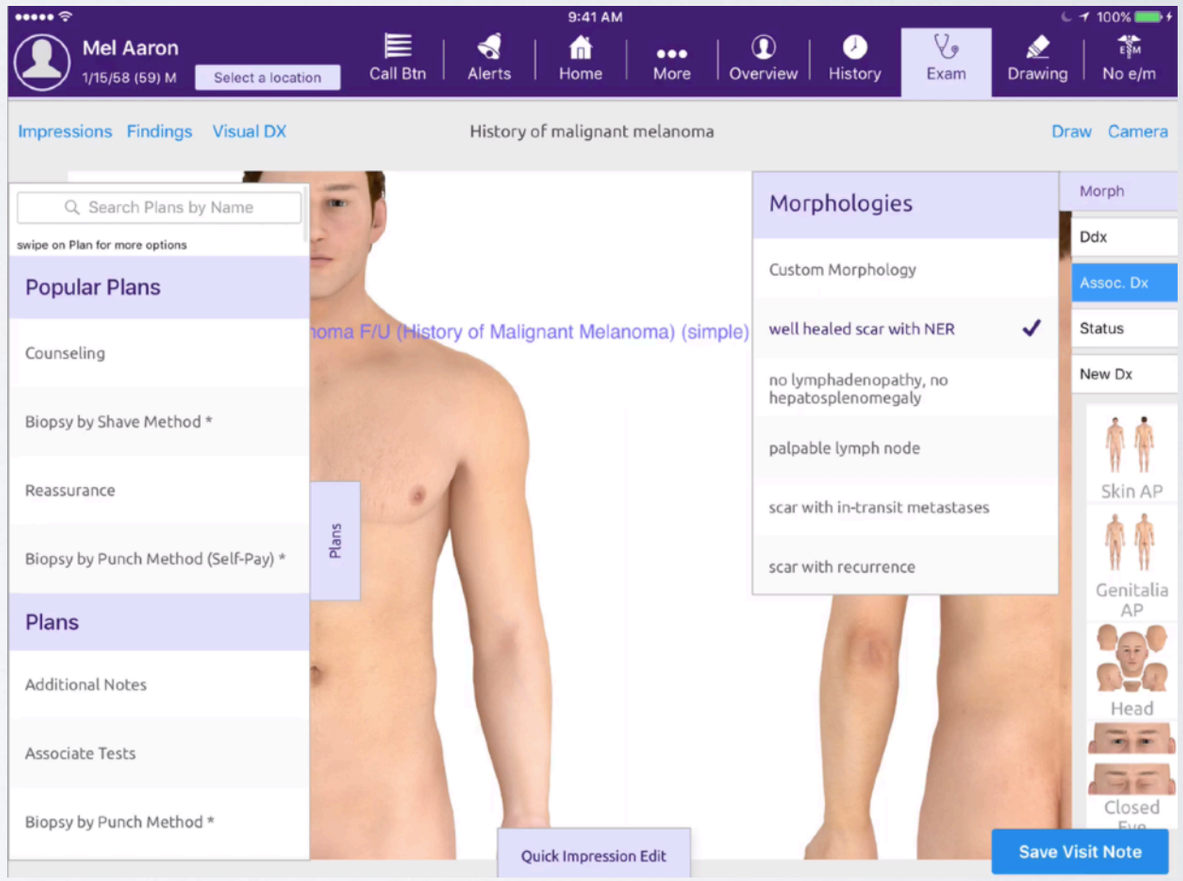


https://www.youtube.com/watch?v=aircAruvnKk&t=25s

http://mccormickml.com/2016/04/19/word2vec-tutorial-the-skip-gram-model/

Example applications

- Predicting sentinel node status in melanoma from a real-world EHR dataset (Richter et al., 2017)
- 30 engineering features / 100,000s of raw features
- 5,000 labeled instances / 2,000,000+ unlabeled instances
- 6 month project, 2 weeks of actual machine learning



http://ieeexplore.ieee.org/document/8217945/

Considerations

- Works best with high-dimensional "raw" data
  - Dimensionality reduction
- Model interpretability can be limited when using a latent feature space
  - Active area of research!

https://www.oreilly.com/ideas/ideas-on-interpreting-machine-learning

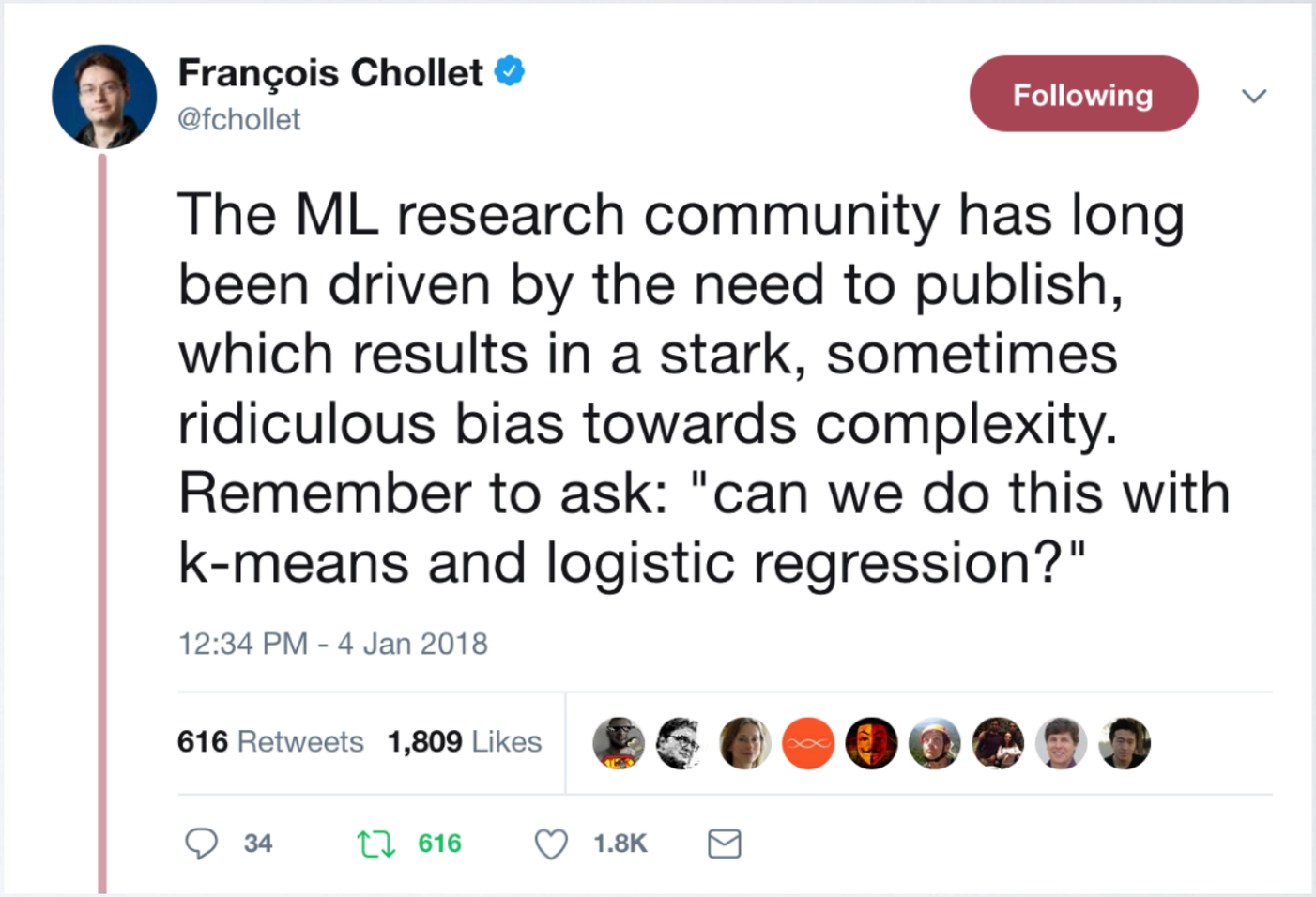
How to do feature learning?

- Unsupervised
  - Matrix factorization
  - Clustering
  - Neural networks (Autoencoders, Restricted Boltzmann machines)
- Supervised
  - Multilayer neural networks (hidden layers create embedding)
  - Transfer learning

https://en.wikipedia.org/wiki/Feature_learning

Aside

- More complexity != better ML
- Sometimes simple methods are good enough
- Don't need deep learning for everything!



https://twitter.com/fchollet/status/948970872802443264

Matrix factorization

- (exactly what it sounds like)
- Also known as matrix decomposition
- Very efficient methods of dimensionality reduction
- Commonly used in ML
  - Eigendecomposition (spectral decomposition)
  - Singular value decomposition (SVD)

https://en.wikipedia.org/wiki/Matrix_decomposition#Decompositions_based_on_eigenvalues_and_related_concepts

SVD

- Singular Value Decomposition
- One way to factorize a matrix
- Diagonal of **Σ** are the *singular values*

$$\mathbf{M} = \mathbf{U\Sigma V}^*$$

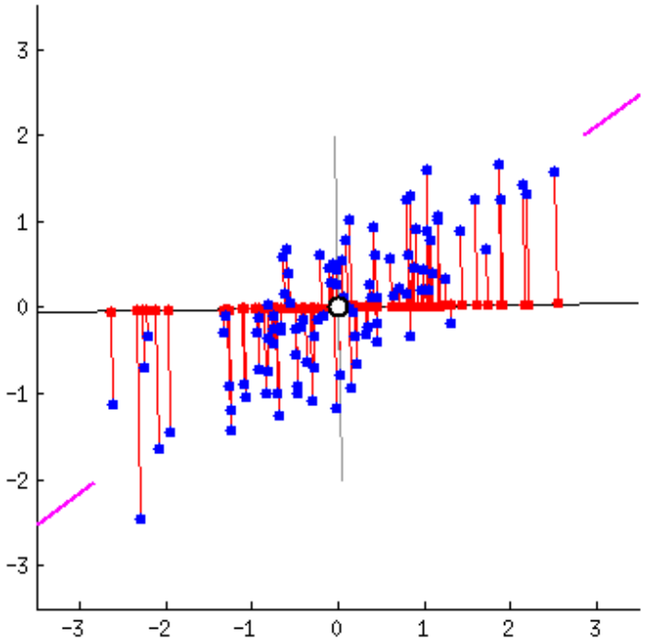
where

- $\mathbf{U}$ is an $m \times m$ unitary matrix over K (if $K = \mathbb{R}$, unitary matrices are orthogonal matrices),
- $\mathbf{\Sigma}$ is a diagonal $m \times n$ matrix with non-negative real numbers on the diagonal,
- $\mathbf{V}$ is an $n \times n$ unitary matrix over K, and
- $\mathbf{V}^*$ is the conjugate transpose of $\mathbf{V}$.

https://en.wikipedia.org/wiki/Singular-value_decomposition

PCA

- Principal Component Analysis
- Find components (new features) that maximize variation between the input features
- Each subsequent principal component is orthogonal to the previous
    - First component captures the most variance



https://stats.stackexchange.com/questions/2691/making-sense-of-principal-component-analysis-eigenvectors-eigenvalues

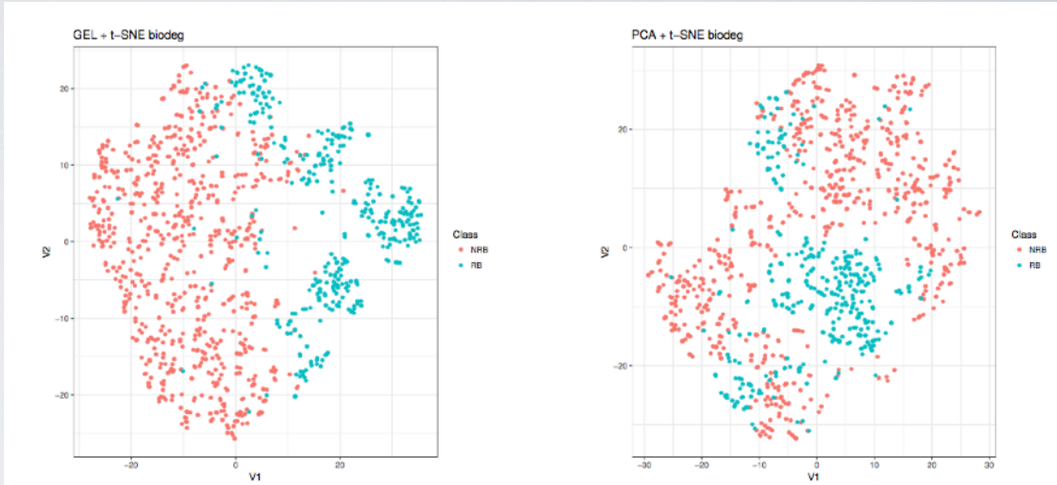
PCA

- Two ways to obtain:
  - (1) Eigenvalue decomposition of correlation matrix
  - (2) SVD of data matrix
- Must center and scale data!
  - Can use Truncated SVD if data is sparse
- General rule of thumb: (1) is more efficient (2) more numerically stable

http://scikit-learn.org/stable/modules/generated/sklearn.decomposition.TruncatedSVD.html
https://stats.stackexchange.com/questions/134282/relationship-between-svd-and-pca-how-to-use-svd-to-perform-pca
https://stats.stackexchange.com/questions/79043/why-pca-of-data-by-means-of-svd-of-the-data

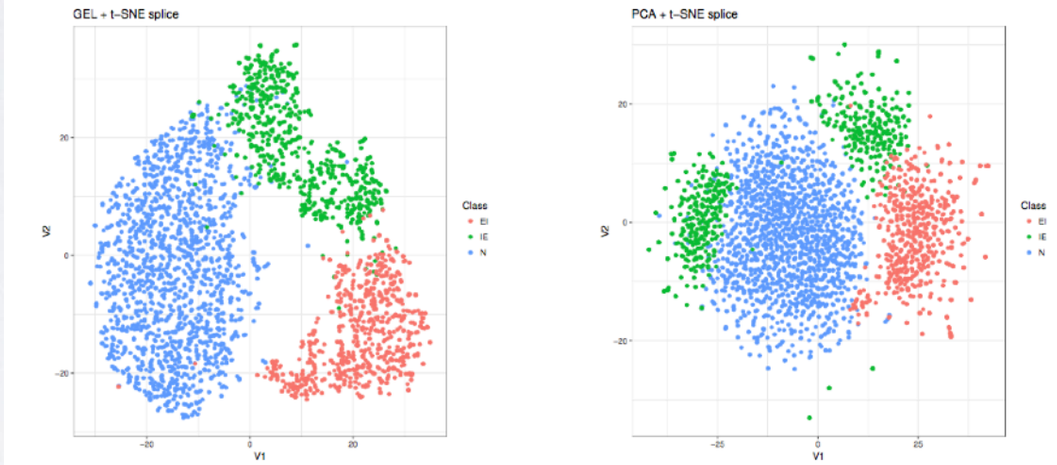
GEL

- Generalized feature embedding learning
- Based on class partitioned instance representation and matrix decomposition
- Can be used in unsupervised or supervised manner
- New algorithm created by Eric Golinko (@egolinko)



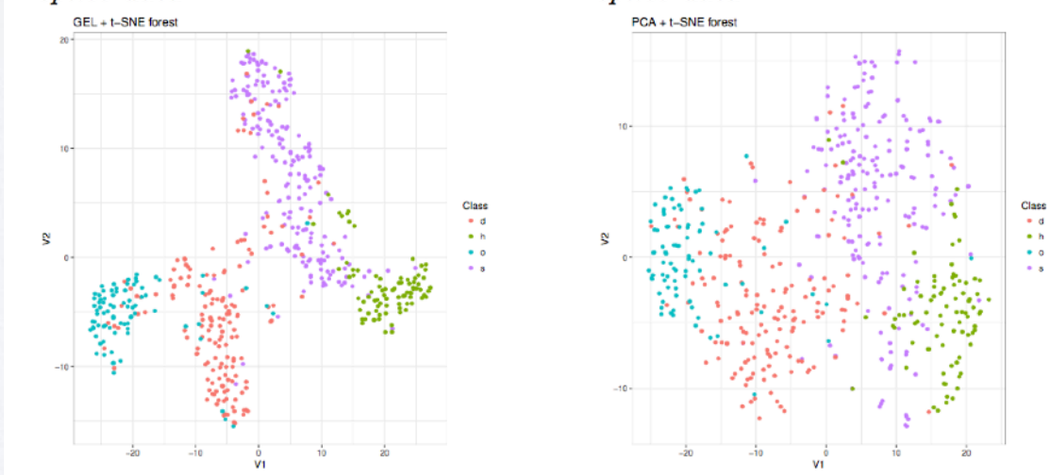
(a) GEL embedding for *biodeg* data

(b) PCA embedding for *biodeg* data



(c) GEL embedding for *splice* data

(d) PCA embedding for *splice* data



(e) GEL embedding for *forest* data

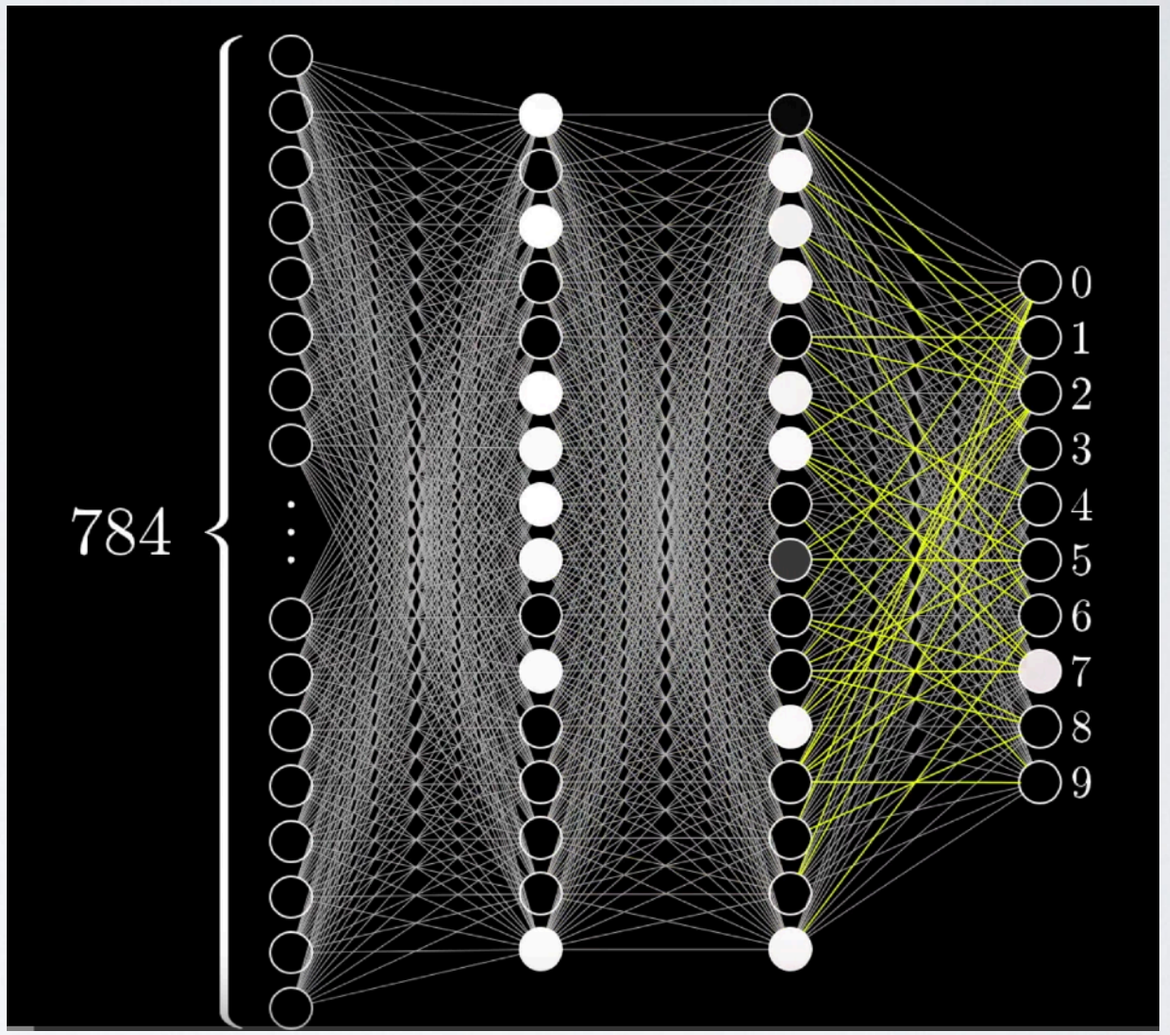
(f) PCA embedding for *forest* data

http://ieeexplore.ieee.org/document/8102942/

https://github.com/egolinko/GEL

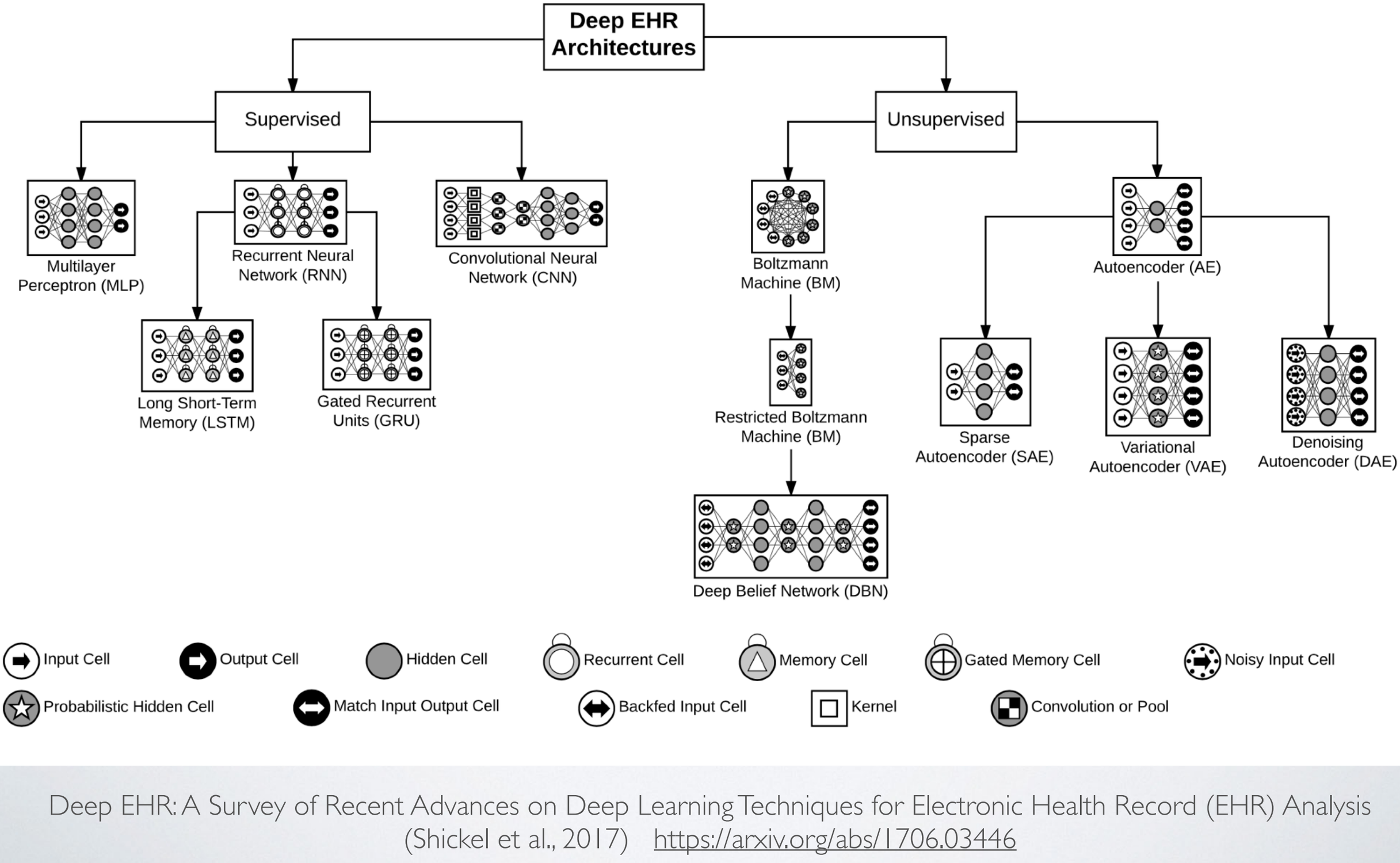
Neural Networks

- Big group of regression functions
- Output of each neuron is based on a non-linear function of inputs
- Learns by minimizing a cost function on the final (output layer) and setting weights through back propagation
- Multiple hidden layers = "deep learning"
- Really great intro to neural networks ->



https://www.youtube.com/watch?v=aircAruvnKk&t=374s

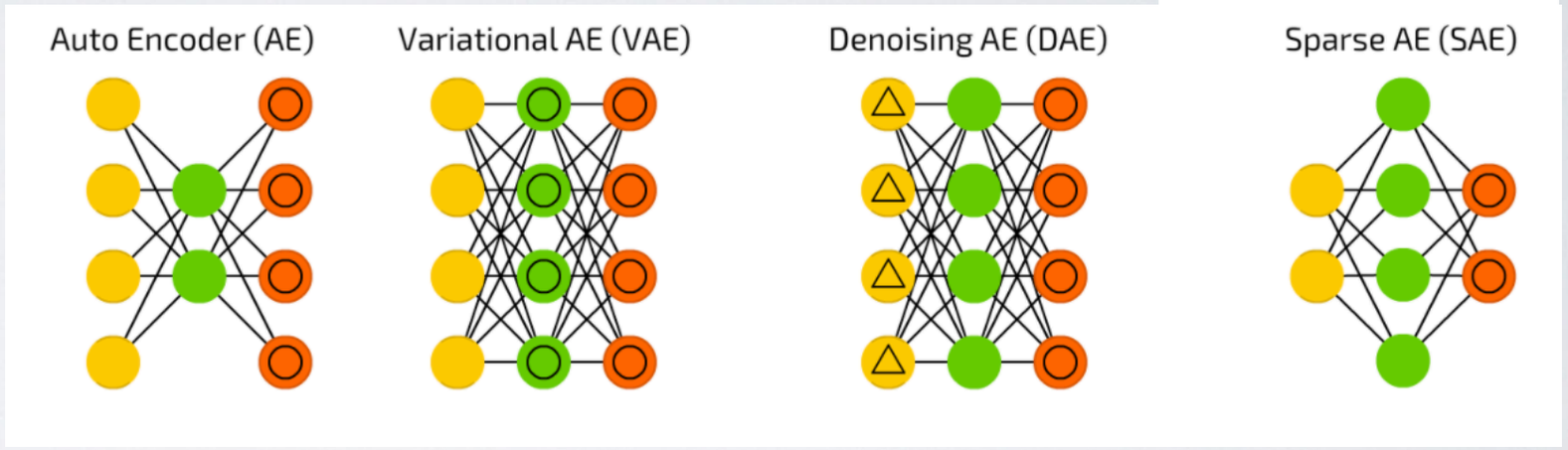
Neural Networks

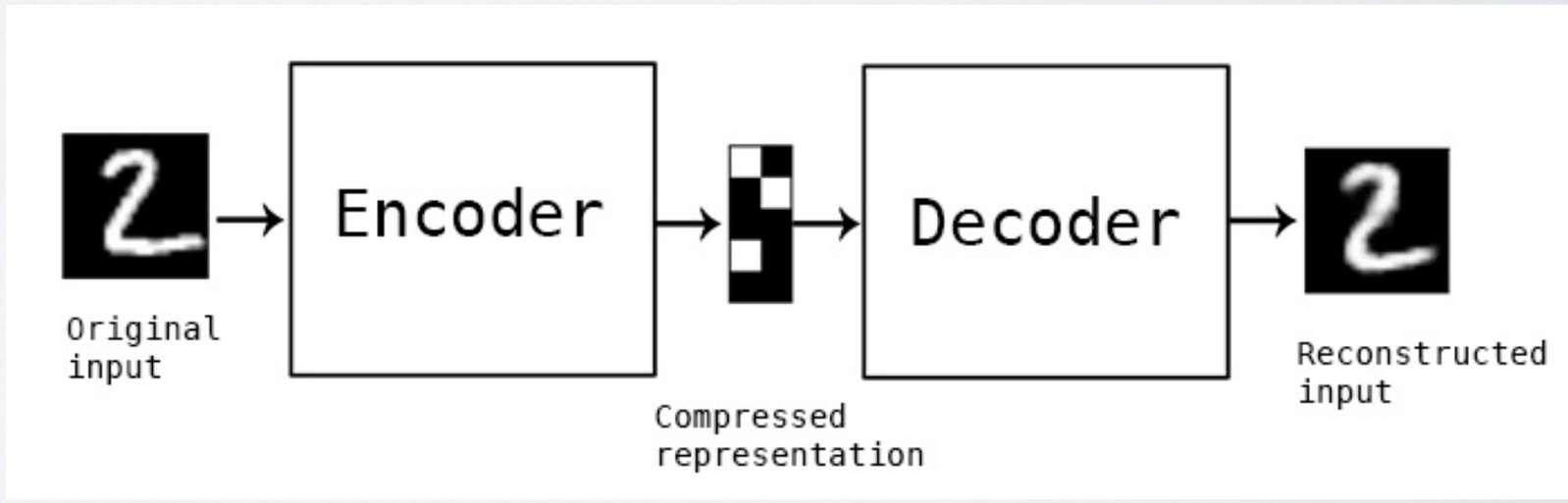


Deep EHR: A Survey of Recent Advances on Deep Learning Techniques for Electronic Health Record (EHR) Analysis (Shickel et al., 2017) https://arxiv.org/abs/1706.03446

Autoencoders

- Reconstruct input
- Use intermediate layer as feature representation
- Useful when you have a large number of unlabeled examples



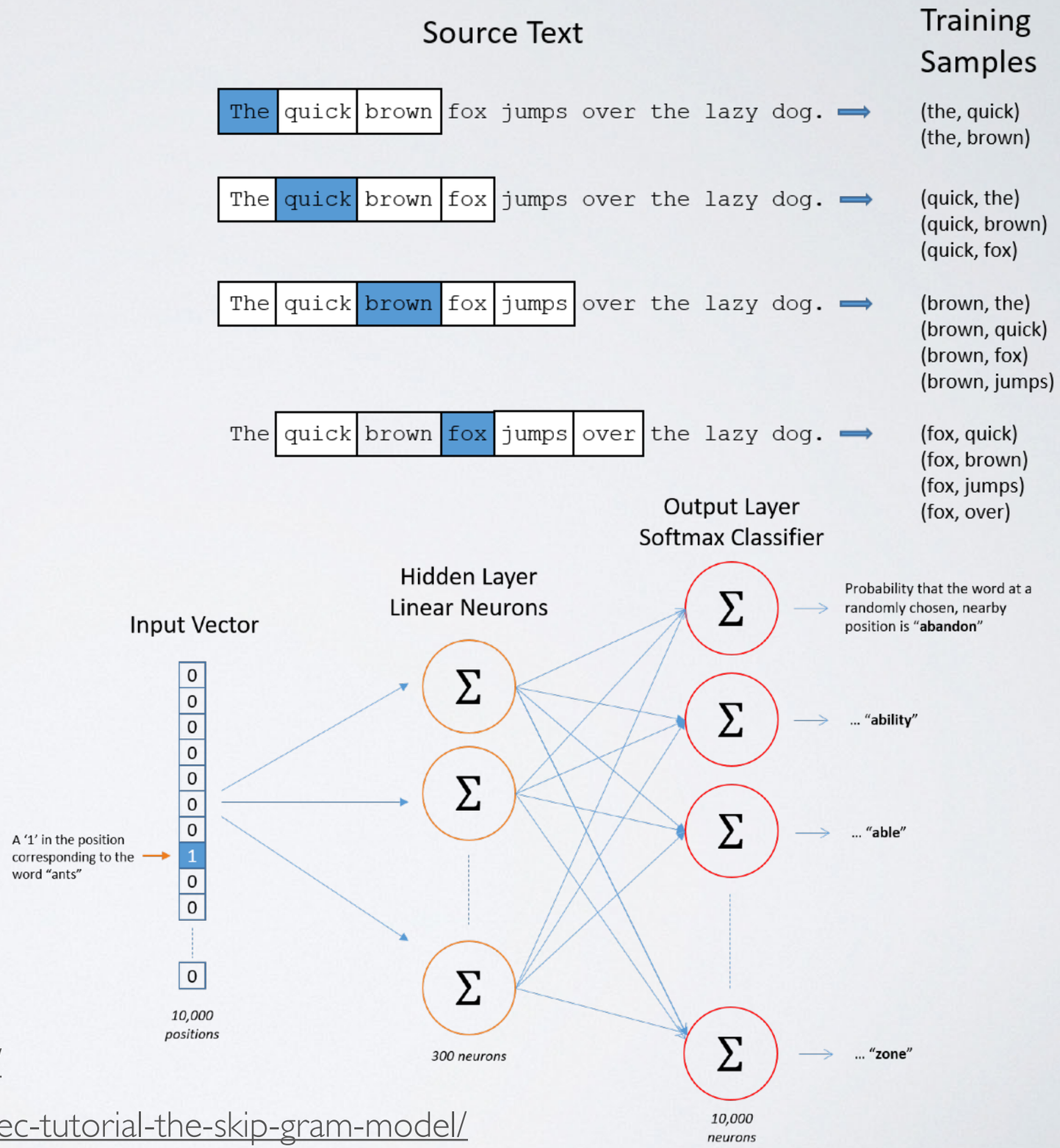


http://www.asimovinstitute.org/neural-network-zoo/

https://blog.keras.io/building-autoencoders-in-keras.html

Skip-gram / Word2Vec

- Model probability that words will be in proximity of a specific word
- Hidden layer creates *word vectors* (embeddings)
- Places words in coordinate space



https://code.google.com/archive/p/word2vec/

http://mccormickml.com/2016/04/19/word2vec-tutorial-the-skip-gram-model/

Supervised + Transfer Learning

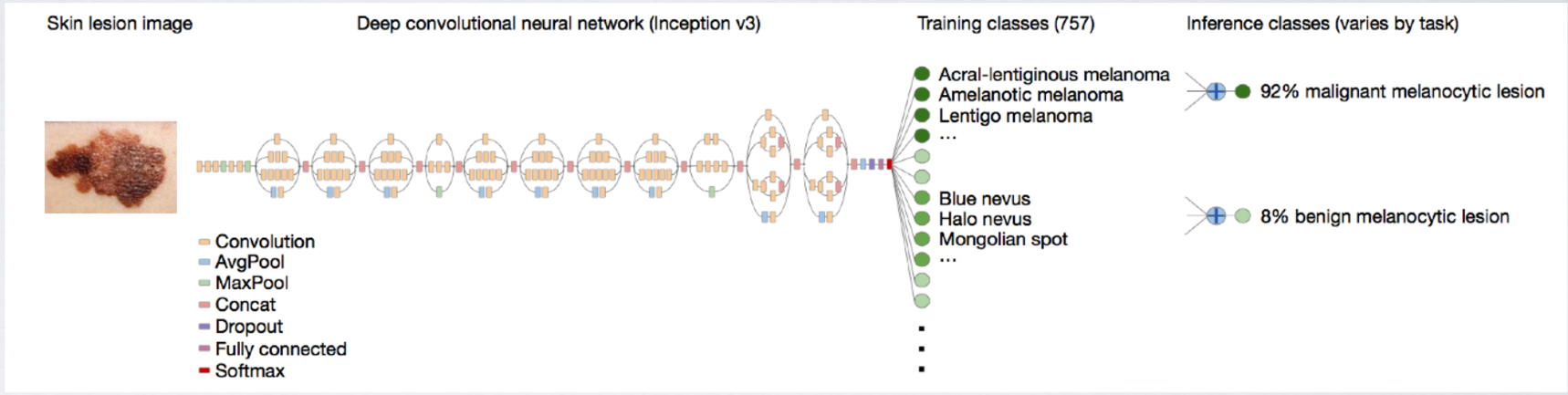
- Train a deep neural network on a supervised learning problem with a large number of labels

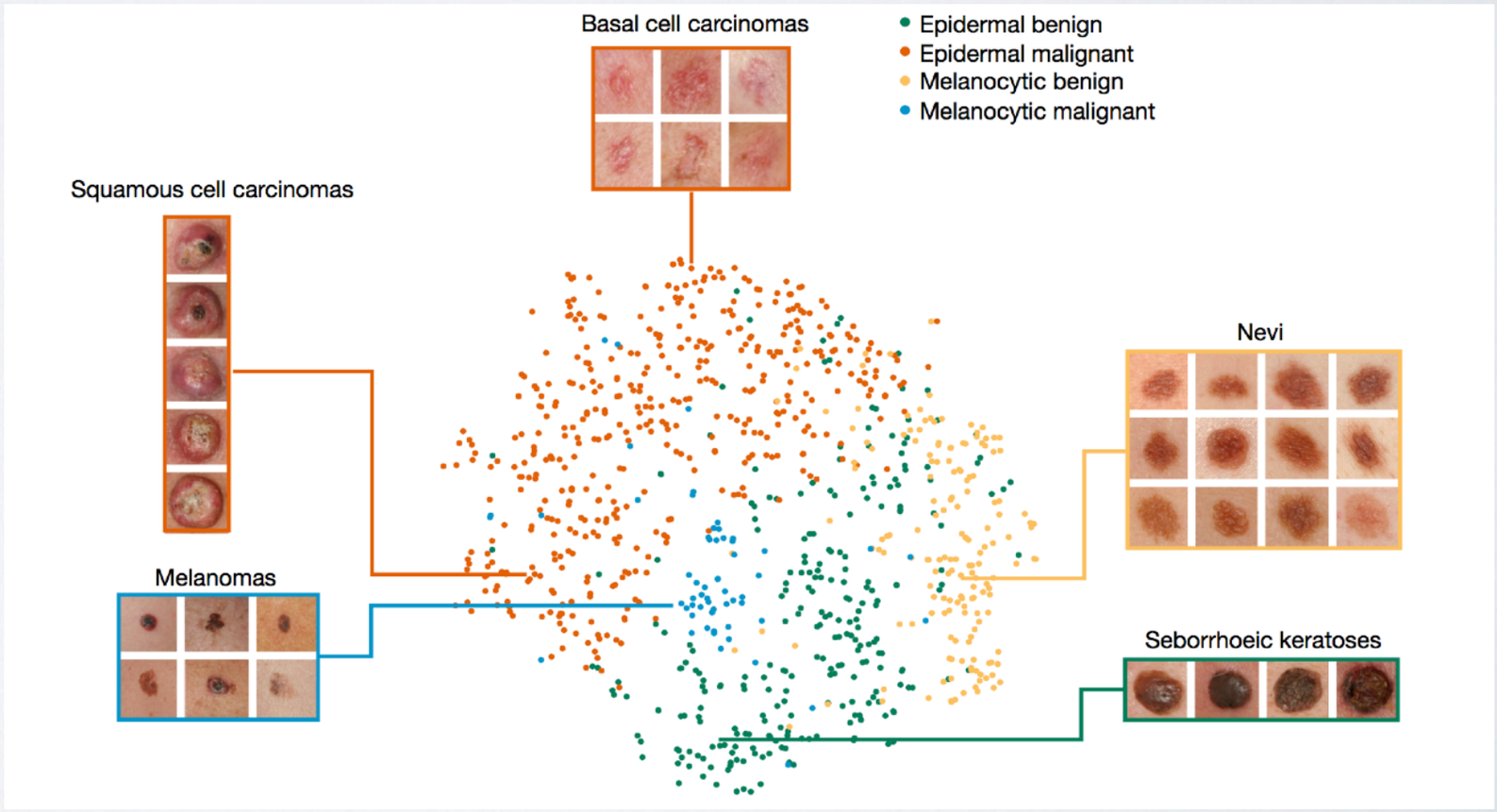
- Chop off output layer, attach new output layer for different dependent variable

- Results in pre-initialized weights, can freeze weights in certain layers when training on new problem

Supervised + Transfer Learning

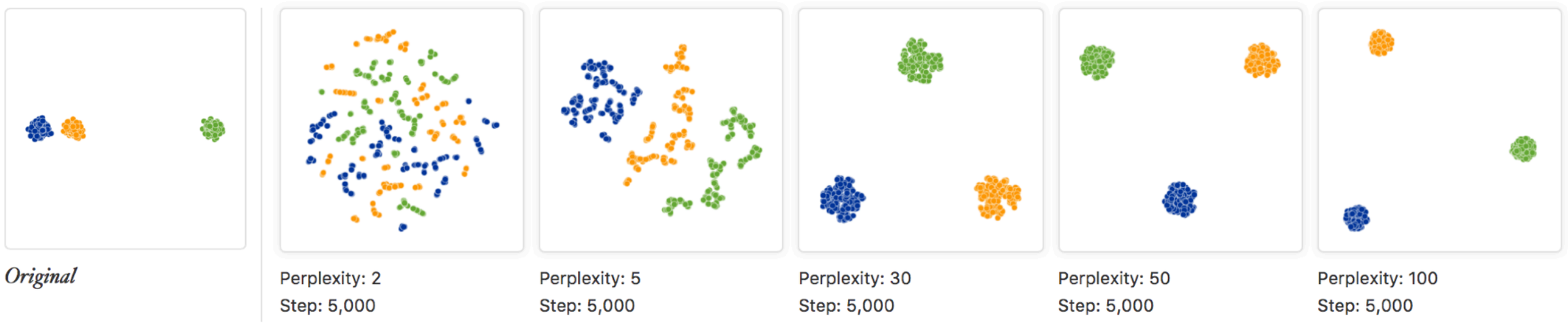
- Dermatologist-level classification of skin cancer with deep neural networks (Esteva et al., 2017)
- Used pre-trained Google Inception V3 CNN pre-trained on ImageNet
- *NOT REPLACING DOCTORS!*





Visualizing feature representations

- t-SNE
- They say it is best used with < 50 features, if more, use PCA first to reduce
    - (I have found it can still work with large feature spaces)
- Don't ask me to explain it :)



https://becominghuman.ai/visualizing-representations-bd9b62447e38

https://distill.pub/2016/misread-tsne/

Thank you!

- Aaron Richter

- Data Scientist

MODERNIZING MEDICINE*

FAU
FLORIDA ATLANTIC UNIVERSITY

- PhD Candidate, Computer Science

- @rikturr rikturr@gmail.com

- *** *We're hiring!!*