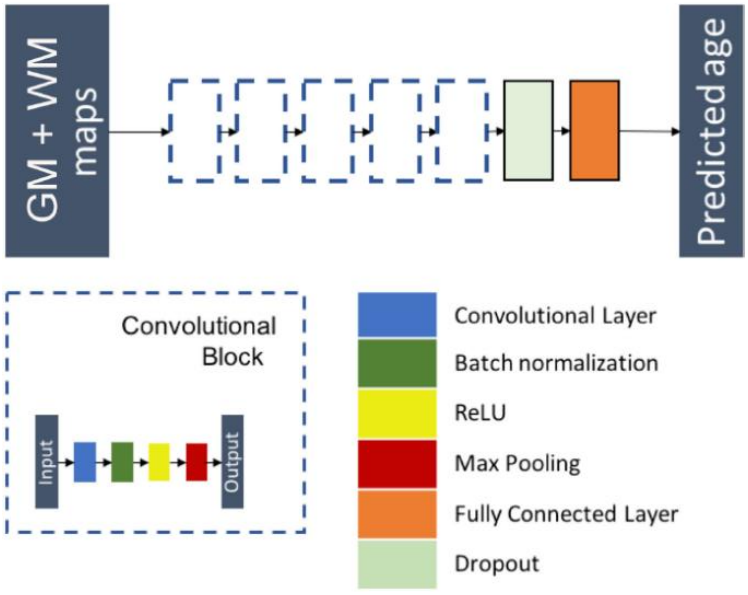
1. Machine learning workflows to estimate class probabilities for precision cancer diagnostics on DNA methylation microarray data
   - **Used Technique / Ingenuity**
     - Models overview
       - 3D maps of gray and/or white matter (deep learning models: six layer CNN, ResNet, and Inception V1)
       - vertex wise measurements from the surface-based processing (models BLUP and SVM)

Model 1	Best Linear Unbiased Predictor(BLUP)
Model 2	Support Vector Regression
Model 3	Six-Layer Convolutional Neural Networks
Model 4	Specialized Six-Layer Convolutional Neural Networks for Younger and Older Subjects
Model 5	ResNet
Model 6	Inception V1

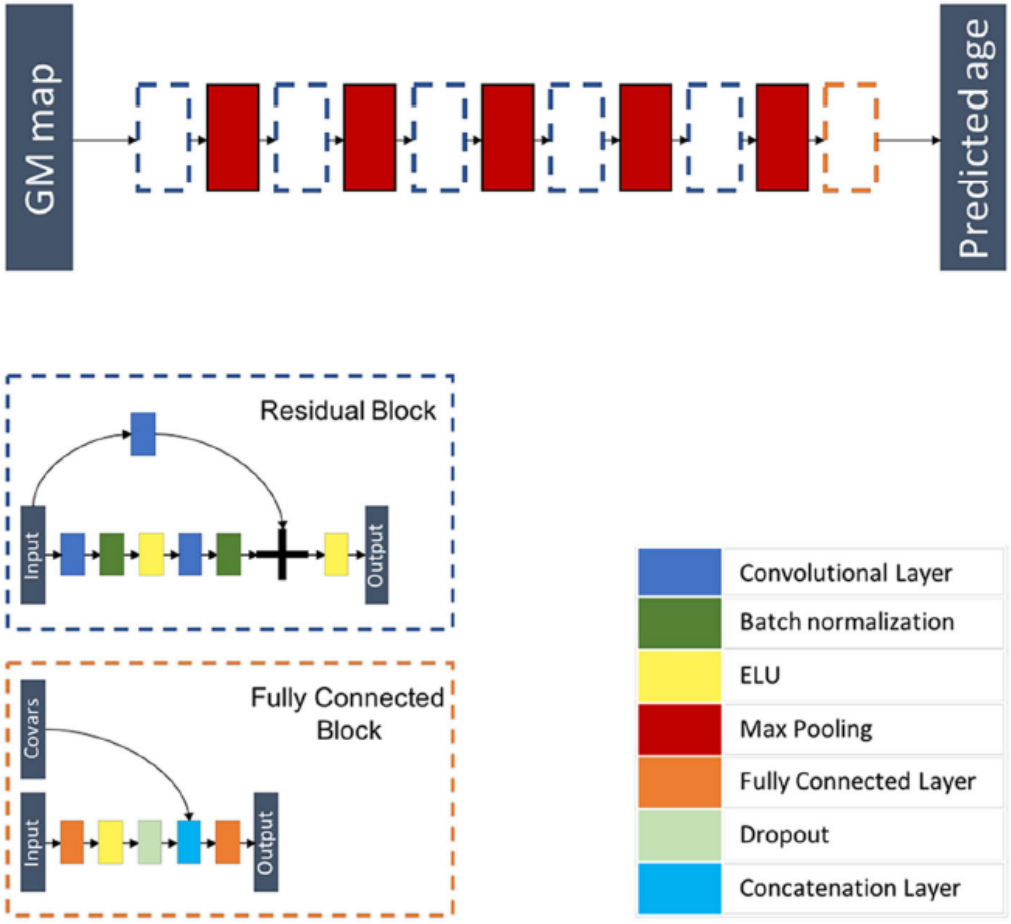
     - Additional Experiments
       - Different Types of Model Combination: Linear Regression vs. Random Forest
       - Combining Seven (Identical) Convolutional Neural Networks or the Seven Best Epochs
       - Influence of the Type of Brain Features on Prediction Accuracy
   - **Suitable Reason**

     In this field, it's very clearly on comparing 6 variety models which can help us to know the implementation what we learned in class.

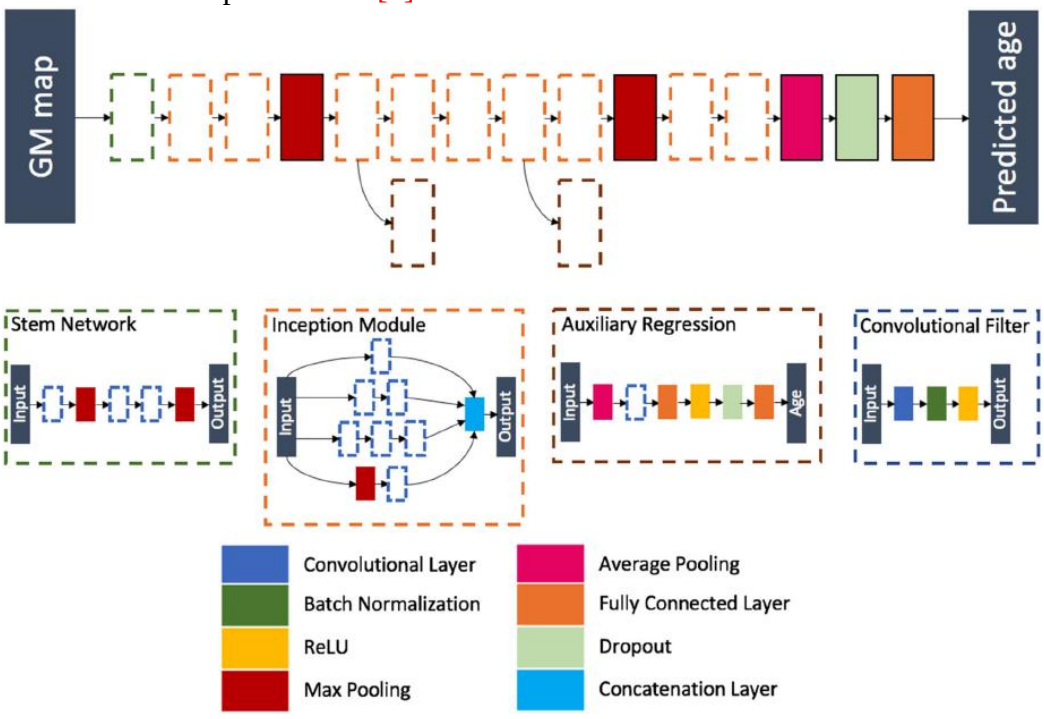
     Also can aware of the result between high level model and custom level model For linear regression and random forest, they trained the **ensemble algorithms** on a random subset. They repeated this process 500 times to get a bootstrap estimate of the SE of the MAE.
   - The custom model in [1]



- Self-defined ResNet in [1]



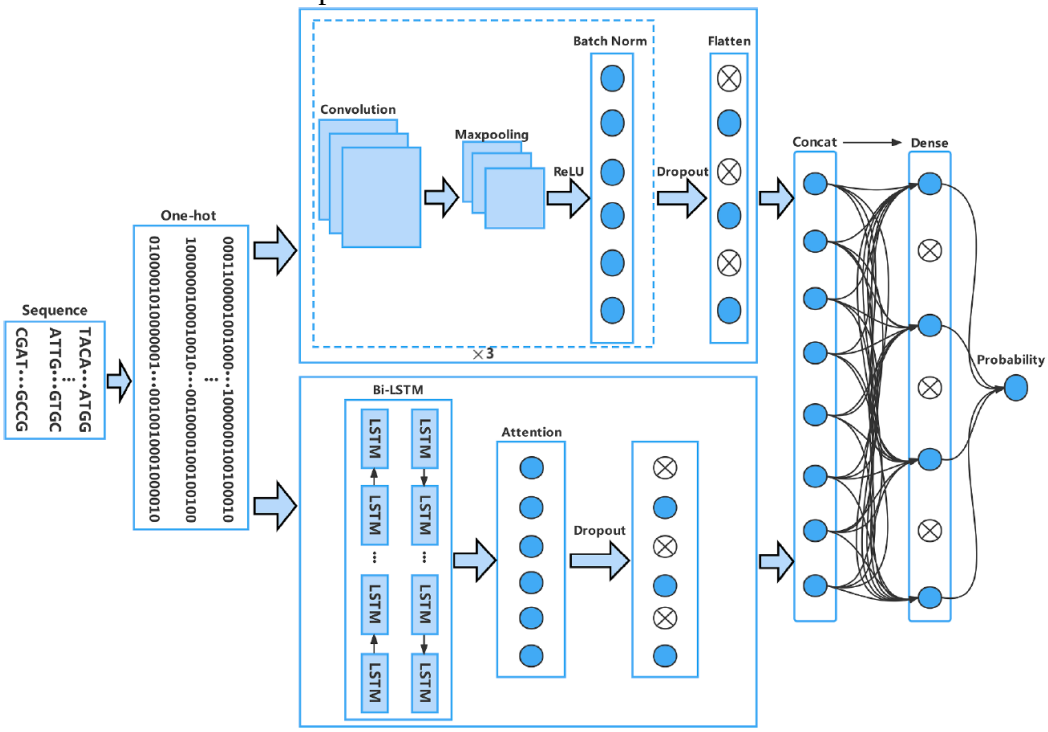
- Self-defined Inception V1 in [1]



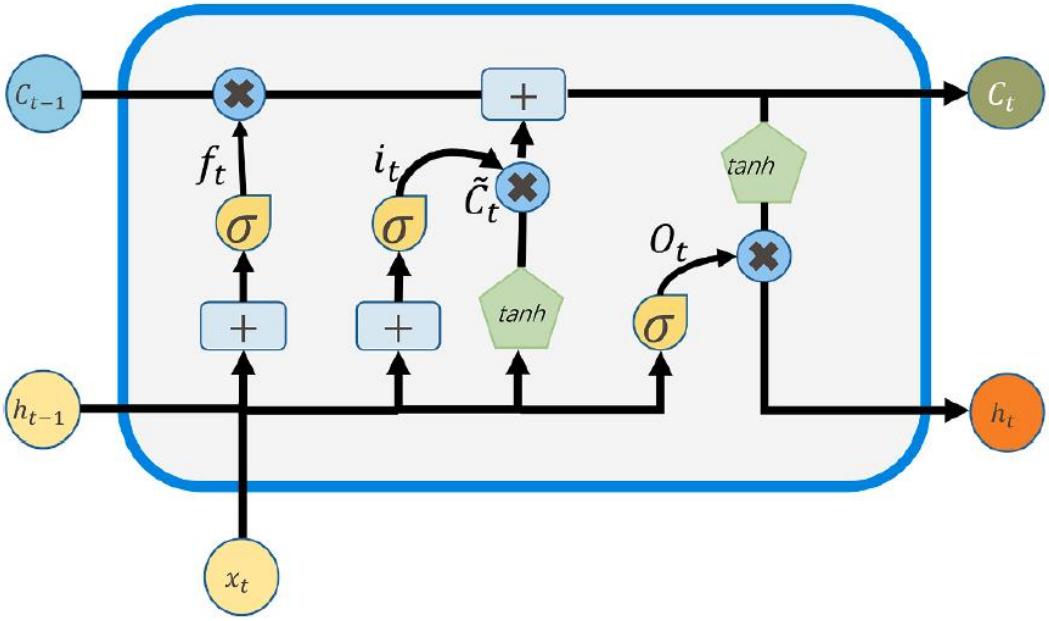
- The whole result of experience in [1]

				Individual algorithms				Ensemble learning			
	BLUP-mean	BLUP-quantiles	SVM	6-layer CNN	Age spe. 6-layer CNN	ResNet	Inception V1	LM	RF	Mean	Median
Fold 1	5.32 (0.19)	4.90 (0.19)	5.31 (0.18)	4.18 (0.16)	4.01 (0.15)	4.02 (0.15)	3.82 (0.14)	3.46 (0.13)*	3.62 (0.15)	3.74 (0.13)	3.67 (0.14)
Fold 2	5.05 (0.18)	4.79 (0.19)	5.34 (0.18)	4.47 (0.15)	4.12 (0.13)	4.01 (0.14)	3.97 (0.15)	3.53 (0.13)*	3.60 (0.15)*	3.69 (0.13)	3.74 (0.13)
Fold 3	4.90 (0.18)	4.37 (0.16)	4.84 (0.17)	4.41 (0.16)	4.27 (0.15)	3.88 (0.14)	4.00 (0.16)	3.33 (0.13)*	3.46 (0.15)*	3.46 (0.12)*	3.45 (0.13)*
Fold 4	5.07 (0.18)	4.71 (0.18)	5.06 (0.18)	4.55 (0.17)	4.27 (0.16)	4.11 (0.15)	3.85 (0.15)	3.57 (0.13)*	3.72 (0.14)	3.68 (0.14)	3.74 (0.15)
Fold 5	5.22 (0.19)	4.69 (0.18)	5.20 (0.18)	4.02 (0.16)	3.89 (0.15)	3.99 (0.16)	3.75 (0.15)	3.34 (0.13)*	3.51 (0.14)	3.56 (0.13)	3.47 (0.13)
5-fold combined MAE	5.11	4.69	5.15	4.33	4.11	4.00	3.88	3.44	3.58	3.62	3.61

2. Deep6mAPred A CNN and Bi-LSTM-based deep learning method for predicting DNA N6-methyladenosine sites across plant species
    - **Used Technique / Ingenuity**
        - The model is shown as below including input, feature extraction and classification
        - **Input**: map a DNA sequence of 41 nt into a binary vector with the one-hot encoding scheme. The feature extraction contained two paralleling parts
        - **Upper part**: mainly of one-dimensional convolution (1D CNN) layer and the batch normalization
        - **Lower part**: bidirectional LSTM (Bi- LSTM) layers and the attention mechanism. The Bi-LSTM was intended to extract contextual semantics of the sequences, while the attention mechanism was to catch the key information
    - The flowchart of the Deep6mAPred



    - The architecture diagram of LSTM



- **Suitable Reason**

  The whole content in this paper has highly connection with Machine Learning Lecture, such as RNN, LSTM, architecture of normal CNN. And they can explained very clearly that why they used this custom model to achieve their goal. Furthermore, it has compared with other models of their performance on 6mA-rice-chen, F.verca and R.chinensis dataset very detailed.

3. Ensemble Learning of Convolutional Neural Network, Support Vector Machine, and Best Linear Unbiased Predictor for Brain Age Prediction

- **Used Technique / Ingenuity**
  - Model for classification
    Random Forest, GLMNet, SVM(including e1071, which is a package of LibSVM in R language, LiblinearR, kernlab, Rgtsvm), and xgboost
  - Calibration Algorithm(i.e. post-processing):logistic regression(GLM function), BRGLM, GLMNet
  - Performance evaluation: HandTill2001
- **Suitable Reason**
  The reason is as the same as [1] which also used various methods and compare it to other papers detailed.
- The reuslt in [3]

Table 2 | Summary table of performance measures of tested classifier algorithms and calibrated workflows

Workflow	Top 10 BS	Classifier	Run-time 5 × 5 CV (/fold)	No. of CPU threads [hardware]	R package	Calibrator	Optimized metric	Hyperparameters	ME	AUC	BS	LL
vRF		RF	38 min	1 [2]	randomForest	raw	ME	ntree =500, mtry=100	0.048	0.999	0.320	0.780
vRF + LR		RF	+30 s (LR)	1 [2]	randomForest	Platt LR (us)	ME	p_{varsel} = 200	0.052	-	0.106	0.289
vRF + LR		RF	+30 s (LR)	1 [2]	randomForest	Platt LR	ME	p_{varsel} = 200	0.052	0.994	0.081	0.262
vRF + FLR		RF	+8–9 min (FLR)	1 [2]	randomForest	Platt Firth (us)	ME	p_{varsel} = 200	0.048	-	0.105	0.193
vRF + FLR		RF	+8–9 min (FLR)	1 [2]	randomForest	Platt Firth	ME	p_{varsel} = 200	0.048	0.999	0.081	0.193
vRF + MR	10	RF	+7–8 min (MR)	11 [2]	randomForest	MR	ME	p_{varsel} = 200	0.043	0.999	0.073	0.155
tRF_{BS}		RF	12–13 h (16–25 min/fold)	72 [5]	randomForest	raw	BS	ntree = (500, 1,000, 1,500, 2,000)	0.055	0.999	0.272	0.673
tRF_{ME}		RF			randomForest	raw	ME	mtry = (80, 90, 100, 110)	0.035	0.999	0.351	0.855
tRF_{LL}		RF			randomForest	raw	LL	p_{varsel} = (100, 200, 500, 1,000,	0.055	0.999	0.273	0.672
tRF_{BS}+LR		RF	+30 s (LR)	1 [2]	randomForest	Platt LR	BS	, 2,000, 5,000, 7,500, 10,000)	0.056	0.997	0.086	0.266
tRF_{ME}+LR	9	RF	+30 s (LR)	1 [2]	randomForest	Platt LR	ME	nodesize = 1	0.042	0.998	0.062	0.156
tRF_{LL} +LR		RF	+30 s (LR)	1 [2]	randomForest	Platt LR	LL	nodesize = 1	0.058	0.995	0.089	0.291
tRF_{BS} +FLR		RF	+8–9 min (FLR)	1 [2]	randomForest	Platt Firth	BS	nodesize = 1	0.054	0.997	0.086	0.194
tRF_{ME} +FLR	8	RF	+8–9 min (FLR)	1 [2]	randomForest	Platt Firth	ME	nodesize = 1	0.037	0.999	0.062	0.150
tRF_{LL} +FLR		RF	+8–9 min (FLR)	1 [2]	randomForest	Platt Firth	LL	nodesize = 1	0.056	0.999	0.089	0.205
tRF_{BS} +MR		RF	+7–8 min (MR)	11 [2]	randomForest	MR	BS	nodesize = 1	0.051	0.997	0.082	0.176
tRF_{ME} +MR	4	RF	+7–8 min (MR)	11 [2]	randomForest	MR	ME	nodesize = 1	0.027	0.999	0.046	0.095
tRF_{LL} +MR		RF	+7–8 min (MR)	11 [2]	randomForest	MR	LL	nodesize = 1	0.055	0.999	0.086	0.188
ELNET (1k)	7	ELNET	~7.5 h (12–15 min/fold)	31 [4]	glmnet	raw	ME	$\alpha = 0$ | 0.025 ; λ = (0.0010–0.0036)	0.032	0.999	0.059	0.131
ELNET (10k)	5	ELNET	~72 h (2–2.25 h/fold)	31 [4]	glmnet	raw	ME	$\alpha = 0$; λ = (0.012–0.038)	0.027	0.999	0.048	0.109
SVM-LK		SVM	~28 h (50–70 min/fold)	11 [3]	e1071	raw	ME	C = 0.001 | 0.01	0.032	0.999	0.372	0.978
SVM-LK+LR	2	SVM	+30 s (LR)	1 [2]	e1071	Platt LR	ME	C = 0.001 | 0.01	0.025	0.999	0.043	0.112
SVM-LK+FLR	3	SVM	+8–9 min (FLR)	1 [2]	e1071	Platt Firth	ME	C = 0.001 | 0.01	0.021	0.999	0.044	0.135
SVM-LK+MR	1	SVM	+7–8 min (MR)	11 [2]	e1071	MR	ME	C = 0.001 | 0.01	0.021	0.999	0.039	0.085
SVM-LK (GPU)	6	SVM	~5 h	1080Ti	Rgtsvm-GPU	global softmax	ME	C = 0.01 | 0.001 ; n.SV= 1,300–1,600	0.033	0.998	0.056	0.144
SVM-CS[6]		SVM	~6 h (13–15 min/fold)	7 [1]	LiblineaR	-	ME	$C \geq$ 0.001	0.028	-	-	-
XGBoost		BT	~65–70 h (110–130 min/fold)	72 [5]	xgboost	raw	ME	Tables 3 and 4	0.051	0.999	0.150	0.430
XGBoost+LR		BT	+30 s (LR)	1 [2]	xgboost	Platt LR	ME	Tables 3 and 4	0.055	0.991	0.087	0.452
XGBoost+FLR		BT	+8–9 min (FLR)	1 [2]	xgboost	Platt Firth	ME	Tables 3 and 4	0.053	0.993	0.089	0.384
XGBoost+MR		BT	+7–8 min (MR)	11 [2]	xgboost	MR	ME	Tables 3 and 4	0.046	0.999	0.092	0.247

Used [hardware]: (i) CPU: [1] 8 threads on i7 7700k at 4.2GHz ; [2] 12 threads on MacBook Pro 15 inches i9-8950HK at 2.9 GHz or [3] i7-6850k at 3.6 GHz; [4] 32 threads on i9-7960X at 2.8 GHz; [5] 72 threads on AWS EC2 c5n.18xlarge at 3.5GHz; (ii) GPU: NVIDIA GTX 1080 Titanium. AUC: multiclass AUC after Hand and Till[70] (that can only be calculated if probabilities are scaled to 1). CS: type 4 without probability output. FLR: 10,000 iterations. XGBoost: using trees as base learners. 1k and 10k: most-variable CpG probes. BT, boosted trees; n.SV, number of support vectors; us, unscaled, rowsum ≠ 1.

A computational solution for bolstering reliability of epigenetic clocks: Implications for clinical trials and longitudinal tracking

Abstract: While using DNA methylation data to make epigenetic clock, the sample variations may lead to large error. Only using the CpGs less noisy (larger variations with noise) cannot solve the problem since it got heavier weight. The PCA method serves as a good solution since it extract the truly important features among those CpGs.

Why suitable: the paper demonstrates a good example for how PCA taught in class can significantly improve the model's quality.

Machine Learning-Based DNA Methylation Score for Fetal Exposure to Maternal Smoking: Development and Validation in Samples Collected from Adolescents and Adults

Abstract: This paper use three data sets(first for model selection and training, second and third for validation) to train a prediction model for fetal exposure to maternal smoking. The paper implements many methods and do training on the first set, choose those performs better and do find tune. Different data selection methods are applied based on each model, and the Imbalance data problem is solved by minority oversampling technique. Finally, they use Cohen's kappa as the criteria for selecting the most suitable model.

Why suitable: the paper provides rich detail and overall process from preprocessing to final evaluation, which can give a good insight to the research field.

Drug Response Prediction Based on 1D Convolutional Neural Network and Attention Mechanism

Abstract: In the paper the team try to develop a model to determine which cancer treatment would be useful for patients through genetic information. First, the data dichotomized into two types for the training label. Next, they identify the drug-sensitive characteristic molecules and their significance index. Then the molecules are used to get the related mRNA, Methyl, and CNV data. These data are used as feature for a 1D CNN combine with attention method.

Why suitable: well-designed feature selection process, a good demonstration of attention algorithm, and a relatively simple model can lead to good solution.

[Hierarchical Ridge Regression for Incorporating Prior Information in Genomic Studies]

基因組數據通常是高維的，高維回歸方法需要正則化(Ridge Regression)，也就是添加迴歸係數的懲罰項。然而，增加此項會需要在模型複雜性（偏差）和模型穩定性（方差）之間的取捨。前人想到可以利用 meta feature 來調整，讓上述問題簡化。

之所以要做出修改，是因為比一般機器學習應用場景 feature 多更多，直接使用係數作為 loss function 會增加太多複雜度，因此使用 meta/ subjective 交互使用的策略降低複雜度，同時減少穩定性的 trade off

這邊提出兩層的 normalization，第一層一樣在 subject-level features 上面做回歸。第二層模擬 meta feature 對 subject-level features 平均值的影響。如此就可以用一般迴歸分析，但又不易產生過於複雜的模型。

[Interpretable machine learning prediction of all-cause mortality]

[eXplainable AI = XAI]

[impact]

在一般主題裡，機器學習的結果就是黑盒。我們無從得知判定的原因。但再生醫領域中，許多判斷會讓病人需要付出很大的成本，甚至是生命，因此希望可以再做決定的時候給病人原因，讓治療過程更為令人安心。這邊使用 gradient boosted tree，產生 tree explainer 來解釋。

文中，他們取得對未來死亡率提供高度信息的風險預測因子，利用靈活的模型捕獲非線性關係(例如: 風險預測因子的“拐點”提供的重要資料)。再利用 xai 重新回推，理解那些 feature 是對死亡重要的。

此篇文章結果比死亡風險評分和生理年齡都更準確，有一定研究價值。

有新的 對死亡重要的原因。

[NetTIME: a multitask and base-pair resolution framework for improved transcription factor binding site prediction]

TF binding prediction 是指給定特定 dna，蛋白質是否會彼此結合，某一特定細胞種類。此題目一直是生醫領域的熱門主題。但一直以都只能用特定 cell type 來預測(因為不同細胞種類情形相差很大)這個模型透過分析 cell type feature 來擬合不同 type 的情況。利用 one-hot encoding 分析，給 auto encoder 做 CNN。

重點可以多放在如何分析 cell type feature (利用幾個 chip -seq 做 feature)這件事情合理性、好不好等等。