How to Win CheXpert Competition?
Deep AUC Maximization in Medical Applications

Zhuoning Yuan

Department of Computer Science
University of Iowa

June 2, 2021

Outline

- Novel Margin-based AUC Loss
- CheXpert Competition
- Melanoma Competition
- LibAUC: A Library for AUC Optimization

- **AUC:** probability of a positive sample ranking higher than a negative sample

$$\begin{aligned}\mathrm{AUC}(\mathbf{w}) &= \Pr(h_{\mathbf{w}}(\mathbf{x}) \geq h_{\mathbf{w}}(\mathbf{x}')|y=1, y'=-1) \\ &= \mathbb{E}\big[\mathbb{I}(h_{\mathbf{w}}(\mathbf{x}) - h_{\mathbf{w}}(\mathbf{x}') \geq 0)\big|y=1, y'=-1\big].\end{aligned}$$

- Convex Surrogate Loss ℓ

$$\min_{\mathbf{w}\in\mathbb{R}^d} \mathbb{E}\big[\ell(h_{\mathbf{w}}(\mathbf{x}) - h_{\mathbf{w}}(\mathbf{x}'))\mathbb{I}_{[y=1]}\mathbb{I}_{[y'=-1]}\big].$$

- Square Loss (existing studies)

$$\ell(h_{\mathbf{w}}(\mathbf{x}) - h_{\mathbf{w}}(\mathbf{x}')) = (1 - h_{\mathbf{w}}(\mathbf{x}) + h_{\mathbf{w}}(\mathbf{x}'))^2$$

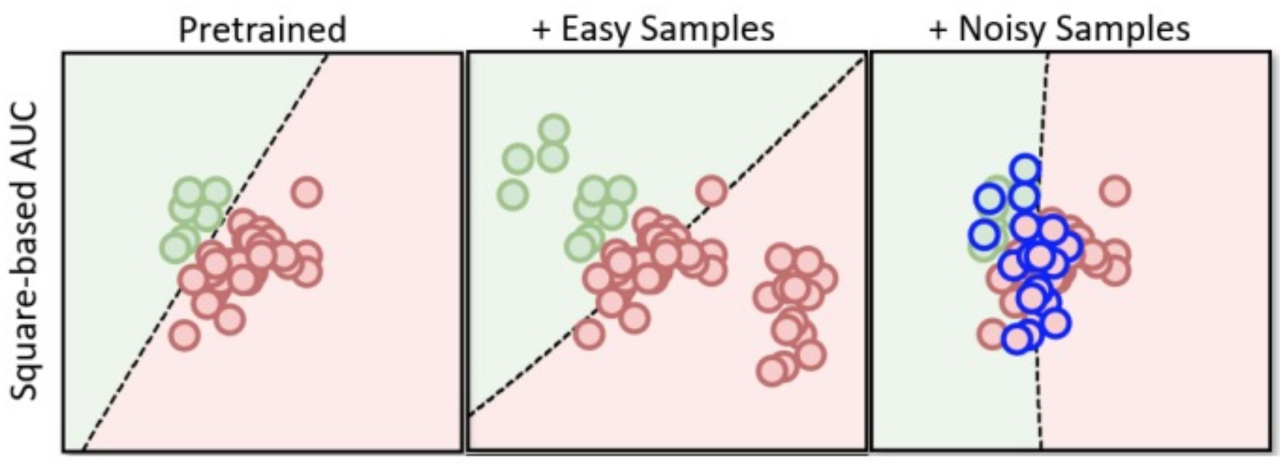
$\Downarrow$ min-max form

$$\min_{\substack{\mathbf{w}\in\mathbb{R}^d \\ (a,b)\in\mathbb{R}^2}} \max_{\alpha\in\mathbb{R}} f(\mathbf{w}, a, b, \alpha) := \mathbb{E}_{\mathbf{z}}\left[F(\mathbf{w}, a, b, \alpha; \mathbf{z})\right],$$

$$\begin{aligned}F(\mathbf{w}, a, b, \alpha; \mathbf{z}) = {} & (1-p)\left(h_{\mathbf{w}}(\mathbf{x}) - a\right)^2 \mathbb{I}_{[y=1]} \\ & + p(h_{\mathbf{w}}(\mathbf{x}) - b)^2 \mathbb{I}_{[y=-1]} - p(1-p)\alpha^2 \\ & + 2\alpha\left(p(1-p) + p h_{\mathbf{w}}(\mathbf{x})\mathbb{I}_{[y=-1]} - (1-p)h_{\mathbf{w}}(\mathbf{x})\mathbb{I}_{[y=1]}\right),\end{aligned} \tag{5}$$

Toy example: Squared-based AUC Loss

- A toy example:
  - 2-layer NN
  - imbalanced 2D data points
  - Adding Easy & Noisy examples
- Performance on Square-based AUC Loss
  - Adverse Effect on Easy Data
  - Sensitive to Noisy Data



Outline

- **Novel Margin-based AUC Loss**
- CheXpert Competition
- Melanoma Competition
- LibAUC: A Library for AUC Optimization

Novel Margin-based AUC Loss

- Decomposition of Square-based AUC Loss

$$A(\mathbf{w}) = \mathrm{E}[(h_{\mathbf{w}}(\mathbf{x}) - a(\mathbf{w}))^2 | y = 1] + \mathrm{E}[(h_{\mathbf{w}}(\mathbf{x}') - b(\mathbf{w}))^2 | y' = 1] + (1 + b(\mathbf{w}) - a(\mathbf{w}))^2$$

$a(\mathbf{w})$ ($b(\mathbf{w})$): average score of positive data (negative data)

- Margin-based AUC Loss:

$$A_1(\mathbf{w}) = \mathrm{E}[(h_{\mathbf{w}}(\mathbf{x}) - a(\mathbf{w}))^2 | y = 1] + \mathrm{E}[(h_{\mathbf{w}}(\mathbf{x}') - b(\mathbf{w}))^2 | y' = 1] + (m + b(\mathbf{w}) - a(\mathbf{w}))_+^2$$

where $[s]_+ = \max(0, s)$, m is a margin parameter.

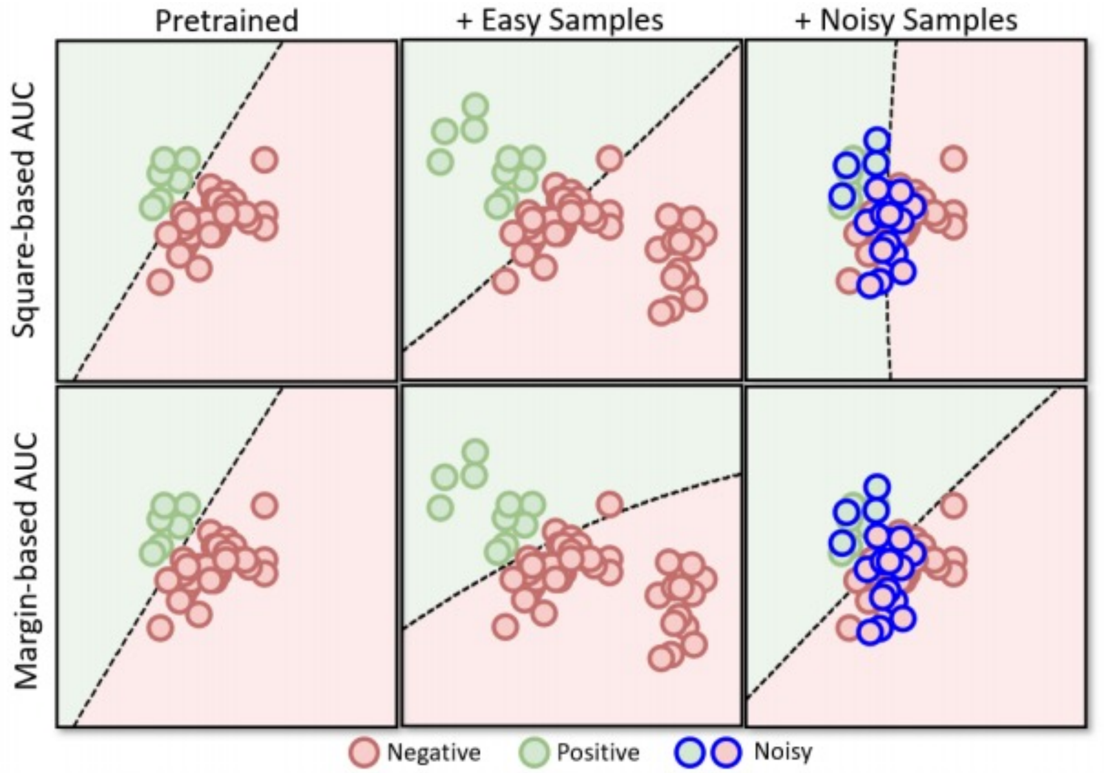
min-max form

$$\min_{\substack{\mathbf{w} \in \mathbb{R}^d \\ (a,b) \in \mathbb{R}^2}} \max_{\alpha \geq 0} \mathbb{E}_{\mathbf{z}} \left[F_M(\mathbf{w}, a, b, \alpha; \mathbf{z}) \right], \quad \textit{where} \tag{8}$$

$$F_M(\mathbf{w}, a, b, \alpha; \mathbf{z}) = (1-p)\left(h_{\mathbf{w}}(\mathbf{x}) - a\right)^2 \mathbb{I}_{[y=1]} \tag{9}$$
$$+ p(h_{\mathbf{w}}(\mathbf{x}) - b)^2 \mathbb{I}_{[y=-1]} - p(1-p)\alpha^2$$
$$+ 2\alpha \left(p(1-p)m + p h_{\mathbf{w}}(\mathbf{x}) \mathbb{I}_{[y=-1]} - (1-p) h_{\mathbf{w}}(\mathbf{x}) \mathbb{I}_{[y=1]} \right).$$

Novel Margin-based AUC Loss

- AUC-margin Loss
  - Robust on Easy Data
  - Robust to Noisy Data



Optimization for Solving AUCM Loss

- Proximal Epoch Stochastic Gradient (**PESG**)
  - ***F_M*** is objective function (AUC-margin loss)
  - ***v*** = (***w, a, b***)
  - ***λ*** is weight decay
  - Stagewise training, i.e., decaying ***η*** **,** resetting ***V_{ref}***

Algorithm 1 PESG for solving AUC margin loss

Require: η, γ, λ, T

1: Initialize $\mathbf{v}_1, \alpha_1 \geq 0$
2: **for** $t = 1, \ldots, T$ **do**
3: Compute $\nabla_{\mathbf{v}} F_{\mathrm{M}}(\mathbf{v}_t, \alpha_t; \mathbf{z}_t)$ and $\nabla_{\alpha} F_{\mathrm{M}}(\mathbf{v}_t, \alpha_t; \mathbf{z}_t)$.
4: Update primal variables

$$\mathbf{v}_{t+1} = \mathbf{v}_t - \eta(\nabla_{\mathbf{v}} F_{\mathrm{M}}(\mathbf{v}_t, \alpha_t; \mathbf{z}_t) + \gamma(\mathbf{v}_t - \mathbf{v}_{\mathrm{ref}})) - \lambda\eta\mathbf{v}_t$$

5: Update $\alpha_{t+1} = [\alpha_t + \eta\nabla_{\alpha} F_{\mathrm{M}}(\mathbf{v}_t, \alpha_t; \mathbf{z}_t)]_+$.
6: Decrease η by a factor and update $\mathbf{v}_{\mathrm{ref}}$ periodically
7: **end for**

Experiments: Benchmark Datasets

- **C2, C10, C100, S10** (Binary)
  - training set: randomly remove some positive samples to make it imbalanced
  - testing set: balanced
  - imbalance ratio: 1% (i.e., #positive/#all)

Table 1. Testing AUC on benchmark datasets with imratio=1%.

Dataset	CE	Focal	AUC-S	AUC-M
C2 (D)	0.718±0.018	0.713±0.009	0.803±0.018	**0.809±0.016**
C10 (D)	0.698±0.017	0.700±0.007	0.745±0.010	**0.760±0.006**
S10 (D)	0.641±0.032	0.660±0.027	0.669±0.070	**0.703±0.030**
C100 (D)	0.588±0.011	0.591±0.017	0.607±0.010	**0.614±0.016**
C2 (R)	0.730±0.028	0.724±0.020	0.748±0.007	**0.756±0.017**
C10 (R)	0.690±0.011	0.681±0.011	0.702±0.015	**0.715±0.008**
S10 (R)	0.641±0.021	0.634±0.024	0.645±0.029	**0.659±0.020**
C100 (R)	0.563±0.015	0.565±0.022	0.587±0.017	**0.596±0.016**

Outline

- Novel Margin-based AUC Loss
- **CheXpert Competition**
- Melanoma Competition
- LibAUC: A Library for AUC Optimization

Experiments: 1^{st} **Place** in CheXpert Competition

- CheXpert Competition
  - Organized by Stanford ML Group (Andrew NG)
  - 150+ submissions worldwide, e.g., *Vingroup Big Data Institute, SenseTime, JF HealthCare, HUST, SJTU, Microsoft, Georgia Tech*

the 1st Place

Stanford ML Group (Andrew Ng)
150+ submissions worldwide

Leaderboard

Will your model perform as well as radiologists in detecting different pathologies in chest X-rays?

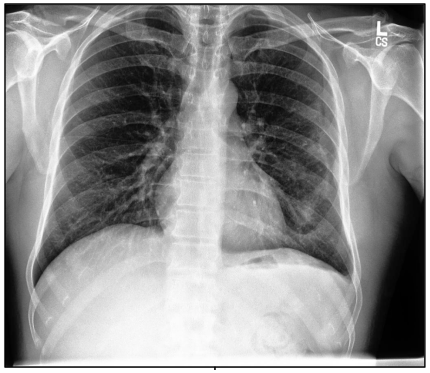
Rank	Date	Model	AUC	Num Rads Below Curve
1	Aug 31, 2020	DeepAUC-v1 *ensemble*	0.930	2.8
2	Sep 01, 2019	Hierarchical-Learning-V1 (ensemble) *Vingroup Big Data Institute* https://arxiv.org/abs/1911.06475	0.930	2.6
3	Oct 15, 2019	Conditional-Training-LSR *ensemble*	0.929	2.6

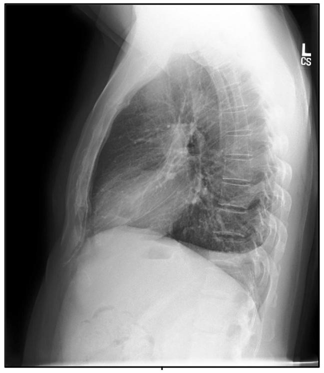
Table: Averaged Testing AUC Scores on CheXpert. NBRC means the # of radiologists out of 3 are beaten by AI algorithms.

Model	AUC	NRBC	Rank
Stanford Baseline (Irvin et al. 2019)	0.9065	1.8	85
YWW (Ye et al. 2020)	0.9289	2.8	5
Hierarchical Learning (Pham et al. 2020)	0.9299	2.6	2
DeepAUC (Ours)	**0.9305**	**2.8**	**1**

CheXpert Competition

- CheXpert is a large-scale X-ray image datasets
  - **224,316** chest radiographs of **65,240** patients
  - Images are labeled for the presence of 14 observations as positive, negative, or uncertain





Pathology	Positive (%)	Uncertain (%)	Negative (%)
No Finding	16627 (8.86)	0 (0.0)	171014 (91.14)
Enlarged Cardiom.	9020 (4.81)	10148 (5.41)	168473 (89.78)
Cardiomegaly	23002 (12.26)	6597 (3.52)	158042 (84.23)
Lung Lesion	6856 (3.65)	1071 (0.57)	179714 (95.78)
Lung Opacity	92669 (49.39)	4341 (2.31)	90631 (48.3)
Edema	48905 (26.06)	11571 (6.17)	127165 (67.77)
Consolidation	12730 (6.78)	23976 (12.78)	150935 (80.44)
Pneumonia	4576 (2.44)	15658 (8.34)	167407 (89.22)
Atelectasis	29333 (15.63)	29377 (15.66)	128931 (68.71)
Pneumothorax	17313 (9.23)	2663 (1.42)	167665 (89.35)
Pleural Effusion	75696 (40.34)	9419 (5.02)	102526 (54.64)
Pleural Other	2441 (1.3)	1771 (0.94)	183429 (97.76)
Fracture	7270 (3.87)	484 (0.26)	179887 (95.87)
Support Devices	105831 (56.4)	898 (0.48)	80912 (43.12)

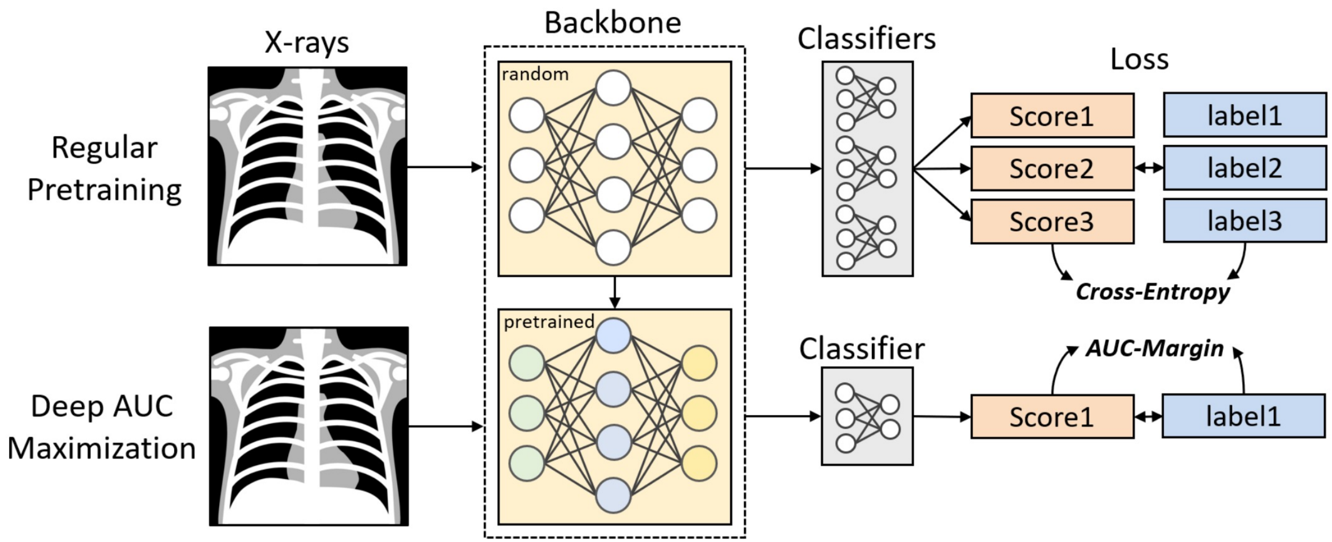
Table 1: The CheXpert dataset consists of 14 labeled observations. We report the number of studies which contain these observations in the training set.

CheXpert Competition

- **Validation Set** contains **200** patients of 234 labeled images
- **Testing Set** contains **500** patients of unknown images which are private for all participants, the leaderboard is evaluated on **5 selected classes:**
  - Atelectasis (肺不张)
  - Cardiomegaly (心脏肥大)
  - Consolidation (肺实变)
  - Edema (肺水肿)
  - Pleural Effusion (胸腔积液)

CheXpert Competition: Our Approach

- **Two-stage learning methods**
  - Feature Learning, i.e., multi-class training using CrossEntropy loss
  - AUC Optimization, i.e., finetuning using AUC optimization



CheXpert Competition: Our Approach

- Model Architectures
  - Family of DenseNet, e.g., DenseNet121, 169, ...
  - InceptionV3
- Image size
  - 224x224, 320x320, 512x512
- Optimization
  - PESG (AUCM)
  - Stagewise Training

Outline

- Novel Margin-based AUC Loss
- CheXpert Competition
- **Melanoma Competition**
- LibAUC: A Library for AUC Optimization

Experiments: Melanoma Competition

- **In-competition**
  - Ranked 33rd out of 3314 teams
  - Ensemble of 10 models
  - Private AUC: **0.9436**
- **Post-Competition**
  - Ensemble of **2 models:** EffecientNetB5 (384x384, AUCM)+ EffecientNetB6 (512x512, CE)
  - Private AUC: **0.9505** (better than winner's 18 ensembled models)

#	△pub	Team Name	Notebook	Team Members	Score	Entries	Last
1	▲ 880	All Data Are Ext			0.9490	116	9mo
2	▲ 55	aloe			0.9485	61	9mo
3	▲ 262	Deloitte Analytics Spain			0.9484	118	10mo

	Public AUC	Private AUC
1st Place	0.9586	0.9490
2nd Place	0.9679	0.9485
DeepAUC (Ours)	0.9559	**0.9505**

Melanoma Competition

- Melanoma is a highly-imbalanced dataset
  - 33,126 training images with only **584** malignant melanoma samples (1.76% positive ratio)
  - 10,982 testing images with **unknown** malignant melanoma samples
- Task
  - to predict malignant or benign (1 or 0)
  - binary classification
- Evaluation
  - AUCROC
  - Public AUC: 30% of testing data
  - Private AUC: 70% of testing data

Melanoma Competition: Our Approach

- **Two-stage learning methods**
  - Feature Learning, i.e., training using CrossEntropy loss
  - AUC Optimization, i.e., finetuning (all layers) using AUC optimization
- Model Architectures
  - EffecientNetB3, B5, B6, B7
- Image Size
  - 256x256, 384x384, 512x512, 768x768
- Optimization
  - PESG (AUCM)
  - Stagewise Learning
  - 5-fold Cross Validation

Melanoma Competition: Our Approach

- Experiments on EffcientNetB5 with 384x384 images
  - TTA (x30) – Testing-time Augmentation
  - Non-TTA

Table: Comparison of Testing AUC on Melanoma dataset for Optimizing EffecientNetB5. TTA (30) means that the results are averaged over 30 times of evaluation on different test-time augmented data.

	wo/ TTA		**w/ TTA(30)**	
Loss	**Public**	**Private**	**Public**	**Private**
CE	0.9391	0.9285	0.9447	0.9345
Focal	0.9412	0.9266	0.9424	0.9303
AUC-S	0.9482	0.9332	0.9502	0.9364
AUC-M	**0.9497**	**0.9357**	**0.9503**	**0.9393**

Other Medical Datasets

- **DDSM:** classification of mammogram for breast cancer screening
  - imratio=13%
  - DenseNet121
- **PCAM:** classification of microscopic images for identifying tumor tissue
  - imratio=1%
  - DenseNet121

Table 5. Testing AUC of two medical datasets on DenseNet121.

Data (imratio)	CE	Focal	AUC-S	AUC-M
DDSM+ (13%)	0.9392	0.9495	0.9469	**0.9544**
PatchCamelyon (1%)	0.8394	0.8556	0.8703	**0.8896**

Outline

- Novel Margin-based AUC Loss
- CheXpert Competition
- Melanoma Competition
- **LibAUC: A Library for AUC Optimization**

LibAUC: Key Features

- **Easy Installation**
  - Easy to install and integrate AUC training pipeline with popular deep learning frameworks like **PyTorch** and **TensorFlow**
- **Large-scale Learning**
  - Robust strategies to handle large-scale optimization on various types of data and make the optimization smoothly
- **Distributed Training**
  - Support for various distributed learning methods that accelerate training efficiency and secure data privacy
- **ML Benchmarks**
  - LibAUC provides a collection of imbalanced classification benchmarks on various applications with easy-to-use input pipeline

LibAUC: A ML Library for AUC Optimization

- **AUROC optimization**
  - Robust Deep AUC optimization
  - Federated Learning
- **AUPRC optimization**
  - **New!**

https://libauc.org/

LibAUC: Demo

- Examples

Installation | Example | Tutorial

```
# install by pip
$ pip install libauc

# check latest version
$ python -c "import libauc; print(libauc.__version_
```

https://github.com/yzhuoning/LibAUC

Get started with python examples

Quickstart for beginners

```
>>> # import library
>>> from libauc.losses import AUCMLoss
>>> from libauc.optimizers import PESG
>>> ...
>>> #define loss
>>> model = model.cuda()
>>> Loss = AUCMLoss()
>>> optimizer = PESG(imratio=0.1)
>>> ...
>>> #training
>>> model.train()
>>> for data, targets in trainloader:
>>>     data, targets  = data.cuda(), targets.cuda()
        preds = model(data)
        loss = Loss(preds, targets)
        optimizer.zero_grad()
        loss.backward(retain_graph=True)
        optimizer.step()
>>> ...
>>> #restart stage
>>> optimizer.update_regularizer()
>>> ...
>>> #evaluation
>>> model.eval()
>>> for data, targets in testloader:
        data, targets  = data.cuda(), targets.cuda()
        preds = model(data)
```

Learn More

LibAUC: Latest Achievement

- **MIT AI Cures Open Challenge for COVID-19**
  - to predict a target compound's property from its molecular structure
  - Collaboration with Prof. **Shuiwang Ji's** group (TAMU)
- Our library (**LibAUC**) helped the team to achieve
  - **1st place** in terms of both **AUROC** and **AUPRC**
- Specifically,
  - **AUROC** improved from 0.925 to 0.957 **(+3%!)**
  - **AUPRC** improved from 0.677 to 0.729 **(+5%!)**

Rank	Model	Author	Submissions	10-fold CV ROC-AUC	10-fold CV PRC-AUC	Test ROC-AUC	Test PRC-AUC
1		DIVE@TAMU	10			0.957	0.729
2	MolecularG	AIDrug@PA	9			0.7	0.725
3		AGL Team	20			0.675	0.702
4		phucdoitoan@Fujitsu	14	0.898 +/- 0.113	0.508 +/- 0.253	0.867	0.694
5	GB	BI	6			0.698	0.67
6	Chemprop ++	AICures@MIT	4			0.877	0.662

Acknowledgements

- **Collaborators**:
  - Zhishuai Guo, Qi Qi, Yan Yan(WSU), Milan Sonka, Shuiwang Ji (TAMU)
- **Supervisor**:
  - Tianbao Yang

Thank You!

Reference

- Advanced Graph and Sequence Neural Networks for Molecular Property Prediction and Drug Discovery
- CheXpert: A Large Chest Radiograph Dataset with Uncertainty Labels and Expert Comparison
- Robust Deep AUC Maximization: A New Surrogate Loss and Empirical Studies on Medical Image Classification
- Stochastic Optimization of Area Under Precision-Recall Curve for Deep Learning with Provable Convergence
- Fast objective and duality gap convergence for non-convex strongly-concave min-max problems
- A patient-centric dataset of images and metadata for identifying melanomas using clinical context
- Stochastic AUC maximization with deep neural networks