Li Weihao

Fast learning rates for plug-in classifiers

This article is written by [AUDIBERT and TSYBAKOV, 2007], with a focus on classification problem. In this article, author derive the rate for excess risk for plug-in estimator under two kind of assumption: one is standard *complexity assumption* for function class, another is *margin assumption*. The problem setting is following: suppose we have data $(\boldsymbol{X}_1, y_1), \cdots (\boldsymbol{X}_n, y_n)$, we want to provide a decision rule $f : \mathbb{R}^p \to \{0, 1\}$ that attain similar performance compared with Bayes optimal classifier, which is given by $f^* = \mathbf{1}_{\{\eta(\boldsymbol{X}) \geq 1/2\}}$ where $\eta(\boldsymbol{X}) = P(Y = 1|\boldsymbol{X})$ denote the regression function of Y on $\boldsymbol{X}$, excess risk is defined as

$$\mathcal{E}\left(\hat{f}_n\right) = \mathbf{E}R\left(\hat{f}_n\right) - R\left(f^*\right) = \mathbf{E}\left(|2\eta(X) - 1|\mathbf{1}_{\left\{\hat{f}_n(X) \neq f^*(X)\right\}}\right)$$

where $R(f) := P(Y \neq f(\boldsymbol{X}))$ denotes the misclassification error of decision rule f.

A straightforward way to get $\hat{f}_n$ is to first estimate $\eta(\cdot)$ via any nonparametric regression method(kernel, local polynomial, orthogonal series, etc.) and we get $\hat{\eta}(\cdot)$, an plug-in classifier is defined as $\hat{f}_n^{\mathrm{PI}}(X) = \mathbf{1}_{\{\hat{\eta}_n(X) \geq 1/2\}}$. And then we adopt following relationship to reduce the excess risk to estimation error of $\hat{\eta}$

$$\mathbf{E}R\left(\hat{f}_n^{\mathrm{PI}}\right) - R\left(f^*\right) \leq 2\mathbf{E}\int\left|\hat{\eta}_n(x) - \eta(x)\right| P_X(dx)$$

[classification error → L_1 estimation error] then based on the complexity assumption (CAR) on function class (e.g., Holder class and its control on metric entropy $\mathcal{H}\left(\varepsilon, \Sigma, L_p\right) \leq A_* \varepsilon^{-\rho} \quad \forall \varepsilon > 0$), we can derive that

$$\sup_{P : \eta \in \Sigma} \mathcal{E}\left(\hat{f}_n^{\mathrm{PI}}\right) = O\left(n^{-\beta/(2\beta+d)}\right), \quad n \to \infty$$

One unsatisfying point is that above relationship is potentially not "sharp"(it is actually sharp in certain minimax sense minimax [Yang, 1999]), the argument is that: for plug-in estimator, we actually do not need to well estimate the regression function, it is sufficient to get a better estimate for optimal decision set, i.e., $G^* = \{x : f^*(x) = 1\} = \{x : \eta(x) \geq 1/2\}$. Instead of characterise the smoothness of regression function, now we put some constraint on complexity of decision set (CAD), i.e., $\mathcal{H}\left(\varepsilon, \mathcal{G}, d_\triangle\right) \leq A_* \varepsilon^{-\rho} \quad \forall \varepsilon > 0$, this assumption is suited to the study of empirical risk minimization (ERM), in [Tsybakov, 2004]

$$\sup_{P : G^* \in \mathcal{G}} \mathcal{E}\left(\hat{f}_n^{\mathrm{ERM}}\right) = O\left(n^{-1/2}\right), \quad n \to \infty$$

But under margin assumption or low noise assumption, people may use ERM or penalized ERM to construct estimator that attain fast rates of convergence, that is, rates that are faster than $n^{-1/2}$,[Koltchinskii, 2006, TSYBAKOV and VAN DE GEER, 2005, Tsybakov, 2004], we first introduce the margin assumption (MA):

$$P_X(0 < |\eta(X) - 1/2| \leq t) \leq C_0 t^\alpha \quad \forall t > 0$$

this condition will favor the classification when α increase, when $\alpha = \infty$, we have perfect "separation" for $\eta(\boldsymbol{X})$, that is, η will away from 1/2. With MA, fast classification rates up to n^{-1} are achievable for ERM type classifiers. In particular, for every $0 < \rho < 1$ and $\alpha > 0$ there exist ERM type classifiers $\hat{f}_n^{\mathrm{ERM}}$ such that

$$\sup_{P : (\mathrm{CAD}), (\mathrm{MA})} \mathcal{E}\left(\hat{f}_n^{\mathrm{ERM}}\right) = O\left(n^{-(1+\alpha)/(2+\alpha+\alpha\rho)}\right), \quad n \to \infty,$$

Compare above result on rate of convergence for plug-in estimator and ERM type estimator, we may conclude that plug-in estimator are generally slow, but this article show that this is wrong by proving plug-in estimator can also achieve fast rate:

Theorem 1 (Thm 3.3 in paper)**.** *Let $\mathcal{P}_\Sigma$ denote the class of all probability distributions P on $\mathcal{Z}$ such that:*
(i) the margin Assumption (MA) is satisfied,
(ii) the regression function η belongs to the Hlder class $\Sigma\left(\beta, L, \mathbf{R}^d\right)$,
(iii) the strong density assumption on P_X is satisfied.

With local polynomial estimator for regression, for any $n \geq 1$ the excess risk of the plug-in classifier $\hat{f}_n^* = \mathbf{1}_{\{\hat{\eta}_n^* \geq 1/2\}}$ with bandwidth $h = n^{-1/(2\beta+d)}$ satisfies

$$\sup_{P \in \mathcal{P}_\Sigma} \left\{\mathbf{E} R\left(\hat{f}_n^*\right) - R\left(f^*\right)\right\} \leq C n^{-\beta(1+\alpha)/(2\beta+d)},$$

For $\alpha\boldsymbol{\beta} > d/2$, it is faster than $n^{-1/2}$
For $\alpha\boldsymbol{\beta} > d$, it is faster than n^{-1}, but in this case $\mathcal{P}_\Sigma$ is very small, only hold for very particular joint distribution of $(\boldsymbol{X}, Y)$

For lower bound, it gives same rate under condition $\alpha\beta \leq d$. For special case $\alpha = \infty$, faster rate can be expected. Suppose there exists $t_0 > 0$ such that $P_X\left(0 < |\eta(X) - 1/2| \leq t_0\right) = 0$. Then

Proposition 2 (Prop 3.7 in paper)**.** *There exists a fixed (independent of n) $h > 0$ such that for any $n \geq 1$ the excess risk of the plug-in classifier $\hat{f}_n^* = \mathbf{1}_{\{\hat{\eta}_n^* \geq 1/2\}}$ with bandwidth h satisfies*

$$\sup_{P \in \mathcal{P}_{\Sigma,\infty}} \left\{\mathbf{E} R\left(\hat{f}_n^*\right) - R\left(f^*\right)\right\} \leq C_4 \exp\left(-C_5 n\right)$$

Some proof idea and technical notes:
1. To prove the main theorem, they first derive a general lemma : some positive sequence a_n, for $n \geq 1$ and any $\delta > 0$, and for almost all x w.r.t. P_X, we have

$$\sup_{P \in \mathcal{P}} P^{\otimes n}\left(\left|\hat{\eta}_n(x) - \eta(x)\right| \geq \delta\right) \leq C_1 \exp\left(-C_2 a_n \delta^2\right)$$

Consider the plug-in classifier $\hat{f}_n = \mathbf{1}_{\{\hat{\eta}_n \geq 1/2\}}$. If all the distributions $P \in \mathcal{P}$ satisfy the margin Assumption (MA), we have

$$\sup_{P \in \mathcal{P}} \left\{\mathbf{E} R\left(\hat{f}_n\right) - R\left(f^*\right)\right\} \leq C a_n^{-(1+\alpha)/2}$$

The first exponential inequality for $\hat{\eta}$ will hold for ranges of nonparametric estimator, for example the author give a example on local polynomial:

$$\sup_{P \in \mathcal{P}} P^{\otimes n}\left(\left|\hat{\eta}_n^*(x) - \eta(x)\right| \geq \delta\right) \leq C_1 \exp\left(-C_2 n^{2\beta/(2\beta+d)} \delta^2\right)$$

2. Relate MA with CAR, we first derive some lemma (5.1 in paper): $R(\bar{f}) - R\left(f^*\right) \leq 2C_0 \|\bar{\eta} - \eta\|_\infty^{1+\alpha}$, lemma (5.2 in paper) $R(\bar{f}) - R\left(f^*\right) \leq C_1(\alpha, p)\|\bar{\eta} - \eta\|_p^{p(1+\alpha)/(p+\alpha)}$ then we can apply some result on L_∞ and L_p estimation error

[Problem]:

- How to use nonparametric regression method to estimate η based on binary response.
- How ERM works for classification problem, algorithm aspect.

  plug-in estimator: $\hat{G}_n = \{x : \hat{\eta}_n(x) \geq 1/2\}$
  ERM classifiers: $\hat{G}_n = \arg\min_{G \in \mathcal{C}} R_n(G)$, where $R_n(G) = \frac{1}{n}\sum_{i=1}^n I\left(Y_i \neq I\left(X_i \in G\right)\right)$

- Relationship between CAR and MA
- Comparison between L_∞ and $L_{1/2}$ bound for nonparametric regression: extra $\log n$ for infinite norm

Li Weihao

Statistical inference for model parameters in stochastic gradient descent

[Chen et al., 2020] First of all, this is a inference paper, in the page two of the papers, they mention that many existing work focus on 'estimation' aspect without give a reference(remark), this is the motivation for this paper. I guess there are some AOS work on 'estimation' based on the SGD.

SGD is an iterative algorithm

$$x_i = x_{i-1} - \eta_i \nabla f\left(x_{i-1}, \zeta_i\right)$$

where ζ denote the random sample from a probability distribution Π and f is the loss function. This algorithm can output either the last iterate x_n (sample size n) or the average iterate $\bar{x}_n = \frac{1}{n}\sum_{i=1}^{n} x_i$. The method that output $\bar{x}_n$ is referred to as averaged SGD (ASGD). As we can see from the iterate itself, it is of low computational burden since gradient only contains one point, which is independent of sample size. Furthermore, SGD use one-pass over the data, therefore there is no need for SGD to store the whole datasets, this is a benefit of storage, which naturally fit in the online settings.

Another extra finding: Origin of the sandwich formula for asymptotic covariance. In parametric model if we use negative log likelihood function as the loss function, the hessian of $F(x) := \mathbb{E}_{\zeta\sim\Pi} f(x,\zeta)$ at the truth $A = \nabla^2 F\left(x^*\right)$ will match with the expectation of 'gradient square' $S = \mathbb{E}\left(\left[\nabla f\left(x^*, \zeta\right)\right]\left[\nabla f\left(x^*, \zeta\right)\right]^T\right)$. That.is to say $S = A^{-1}$. But when loss function is not negative log likelihood function, asymptotic normality will give a statement (use SGD result as an example [Ruppert, 1988, Polyak and Juditsky, 1992]) similar to those result for estimating equation (estimating equation does not require distribution):

$$\sqrt{n}\left(\bar{x}_n - x^*\right) \Rightarrow \mathcal{N}\left(0, A^{-1}SA^{-1}\right)$$

Since this paper focus on the inference, they aim to construct an estimator for $A^{-1}SA^{-1}$ in an online fashion (without the need of storing the data). They provide two estimator for $A^{-1}SA^{-1}$, one is faster in convergence but need to compute hessian matrix and its inverse (plug-in estimator), another is slow in convergence but has some storage advantage (Batch-means estimator)
This kind of paper structure can be used for other situation, prove an plug-in estimator are consistent estimator and then propose another consistent estimator that are computational cheap and require less storage.

Plug-in estimator A thresholding estimator $\widetilde{A}_n$ of A based on the sample estimate $A_n = \frac{1}{n}\sum_{i=1}^{n} \nabla^2 f\left(x_{i-1}, \zeta_i\right)$. Note that this is not the standard sample estimate since each term $\nabla^2 f\left(x_{i-1}, \zeta_i\right)$ is regarding different SGD iterates x_{i-1} (in contrast to a single $\bar{x}_n$), and thus can be constructed online. Similar construction for S_n, the asymptotic covariance $A^{-1}SA^{-1}$ is estimated by $\widetilde{A}_n^{-1} S_n \widetilde{A}_n^{-1}$, which is proven to be a consistent estimator. This paper focus on fixed dimension situation, dimension dependence is very complicated since lots of constant depends on dimension.

Batch-means estimator The plug-in estimator require computation of Hessian matrix A_n and store it, which is not favoured since built-in tool for SGD does not have this function. The idea of Batch-means estimator is to construct independent batch and then sample covariance suffices. When n is sufficiently large, x_n is closed to x^*, then $\nabla F\left(x_{n-1}\right) \approx \nabla F\left(x^*\right) + \nabla^2 F\left(x^*\right)\left(x_{n-1} - x^*\right) = A\Delta_{n-1}$, based on relationship Eq.(1), we have $\Delta_n \approx \left(I_d - \eta_n A\right)\Delta_{n-1} + \eta_n \xi_n$. For large j and k, the strength of correlation between Δ_j and Δ_k is approximately

$$\prod_{i=j}^{k-1}\left\|I_d - \eta_{i+1}A\right\| \approx \exp\left(-\lambda_{\min}(A)\sum_{i=j}^{k-1}\eta_{i+1}\right)$$

Therefore, the correlations between the batch-means $\bar{X}_{n_k}$ are close to zero if the batch sizes are large enough, in which case different batch-means can be roughly treated as independent.

Background on SGD analysis : We can add a population term and subtract it,

$$x_n = x_{n-1} - \eta_n \nabla F\left(x_{n-1}\right) + \eta_n \xi_n,$$

where $\xi_n := \nabla F\left(x_{n-1}\right) - \nabla f\left(x_{n-1}, \zeta_n\right)$ is a martingale difference sequence under assumption (For other problems, this may not hold since here we estimate model parameter). SGD is driven by population gradient. Let $\Delta_n := x_n - x^*$, we have

$$\Delta_n = \Delta_{n-1} - \eta_n \nabla F\left(x_{n-1}\right) + \eta_n \xi_n, \tag{1}$$

Theory of SGD tell us

$$\sqrt{n} \cdot \bar{\Delta}_n \Rightarrow \mathcal{N}\left(0, A^{-1} S A^{-1}\right) \quad \text{if } \alpha \in \left(\frac{1}{2}, 1\right)$$

Li Weihao

Online Sufficient Dimension Reduction Through Sliced Inverse Regression

[Cai et al., 2020] consider estimate the central space in an online fashion, they propose an online updating for kernel matrix, which is not hard to understand (rank one updating), the problem is that they need to store the kernel matrix that take $O(p^2)$ memory, which p is large, $O(p^2)$ can be a problem. Given the stored kernel matrix, they propose 1. online EVD with SGD 2. online EVD with perturbation method to update the eigenspace. For both online strategies, they establish the almost sure convergence for estimated column space.

Li Weihao

Sparse generalized eigenvalue problem

- algorithms based on a convex formulation, for example the Fantope projection and selection (FPS), overcome this difficulty, but are computationally expensive.
- For sprase PCA/CCA, many papers does not have real data analysis, it is intuitive true because they are unsupervised learning and hard to measure goodness in real data example.

Li Weihao

Princeton ORFE Deep Learning Theory Summer School

Video, Misha Belkin

- USLLN+ capacity control ⇒ Generalization of ERM solution

$$E\left(L\left(f_{ERM}^{*}, y\right)\right) \leq \frac{1}{n} \sum L\left(f_{ERM}^{*}\left(x_{i}\right), y_{i}\right)+O^{*}\left(\sqrt{\frac{c(X)}{n}}\right)$$

- Laplace kernel is preferred over Gaussian kernel. Empirical studies suggest that Laplace is very robust.
- Gaussian kernel is essentially linear classifier if noise level is very high. When noise level is very low, Gaussian kernel is somehow closed to nearest neighbor.

Andrea Montanari : statistical viewpoint of deep learning: introduction and motivation

- $R(f;\mathbb{P})=\mathbb{E}\left\{\left(y_{\text{new }}-f\left(\boldsymbol{x}_{\text{new }}\right)\right)^{2}\right\}, \quad\left(y_{\text{new }}, \boldsymbol{x}_{\text{new }}\right) \sim \mathbb{P}$. In practice, we use ERM

$$\hat{R}_{n}(\boldsymbol{\theta}):=\frac{1}{n} \sum_{i=1}^{n}\left(y_{i}-f\left(\boldsymbol{x}_{i} ; \boldsymbol{\theta}\right)\right)^{2}$$

- gradient flow (ODE) and gradient descent are closed when eps small.
- we want to find a solution to interpolate data, we use the taylor expansion, and if we control and second order derivative, i.e., control the lipschitz constant of gradient, then by fixed point iteration theorem, we can show there exist an interpolator of certain form. Moreover, if the second order derivative is very small, we are actually get a linear function. TO sum up, there exists an interpolator that is well approximated by replacing the nonlinear model by its linearization.
- $$f_{\text{lin }}(\boldsymbol{x} ; \boldsymbol{\theta}):=f\left(\boldsymbol{x} ; \boldsymbol{\theta}_{0}\right)+\left\langle\boldsymbol{\theta}-\boldsymbol{\theta}_{0}, \nabla_{\boldsymbol{\theta}} f\left(\boldsymbol{x} ; \boldsymbol{\theta}_{0}\right)\right\rangle .$$
- this motivate us to investigate linearized version of empirical risk and test error:

$$\begin{aligned}\hat{R}_{\text{lin }, n}(\boldsymbol{\theta}) : &= \frac{1}{n}\left\|\boldsymbol{y}-f_{\text{lin}, n}(\boldsymbol{\theta})\right\|_{2}^{2} \\ &= \frac{1}{n}\left\|\tilde{\boldsymbol{y}}-\boldsymbol{\Phi}\left(\boldsymbol{\theta}-\boldsymbol{\theta}_{0}\right)\right\|_{2}^{2},\end{aligned}$$

$$R_{\text{lin }}(\boldsymbol{\theta}):=\mathbb{E}\left\{\left(y_{\text{new }}-f_{\text{lin }}\left(\boldsymbol{x}_{\text{new }} ; \boldsymbol{\theta}\right)\right)^{2}\right\}$$

Andrea Montanari : statistical viewpoint of deep learning:

- under condition of Lipschitz constant L_n, we can get three conclusion. There are many debates over the assumption when we state such theorem, whether it is satisfied by NN, in his opinion, most of time not satisfied. But some people do some simulation and experimental data suggest that this may not a bad assumption. But start with simple linear model is always a good starting point of the analysis.

Dan Roberts and Sho Yaida :Effective theory of DL: beyond the infinite-width limit

- representation learning: for large NN
- Focus on real deep NN, infinite width are not realistic but a simple place to start with. But infinite-width limit leads to a poor model of DNN in practice, don't have representation learning. The central limiting problem is that the input of an infinite number of signals is such that the leveling law of large numbers completely obscures the subtle correlations between neurons that get amplified over the course of training for representation learning.
- simplicity of infinite width:
  - can only focus on the first derivative of Taylor expansion, and it is irrevelant to the algorithm we use because training dynamics will be linear in this limit

Andrea Montanari : Third lecture focus on the work: Surprises in High-Dimensional Ridgeless Least Squares Interpolation. In this work, they consider the linear regression model

$$y_i = \langle \boldsymbol{\beta}_*, \boldsymbol{z}_i \rangle + w_i, \quad \mathbb{E}\left(w_i \mid \boldsymbol{z}_i\right) = 0, \quad \mathbb{E}\left(w_i^2 \mid \boldsymbol{z}_i\right) = \tau^2$$

$\boldsymbol{z}_i \sim \mathrm{N}(0, \boldsymbol{\Sigma}) \perp w_i \sim \mathrm{N}\left(0, \tau^2\right)$ and analyze the excess test wrror of ridge solution

$$\hat{\boldsymbol{\beta}}(\lambda) := \arg \min_{\boldsymbol{b} \in \mathbb{R}^p} \left\{ \frac{1}{n} \|\boldsymbol{y} - \boldsymbol{Z}\boldsymbol{b}\|_2^2 + \lambda \|\boldsymbol{b}\|_2^2 \right\}$$

where excess test error is the difference between test error and Bayes error, defined as

$$\begin{aligned} R_{\text{exc}}\left(\lambda; \boldsymbol{Z}, \boldsymbol{\beta}_0, \boldsymbol{w}\right) &:= \mathbb{E}_{\boldsymbol{z}_{\text{new}}} \left\{ \left(\left\langle \hat{\boldsymbol{\beta}}(\lambda), \boldsymbol{z}_{\text{new}} \right\rangle - \left\langle \boldsymbol{\beta}_*, \boldsymbol{z}_{\text{new}} \right\rangle \right)^2 \right\} \\ &= \left\| \hat{\boldsymbol{\beta}}(\lambda) - \boldsymbol{\beta}_* \right\|_{\boldsymbol{\Sigma}}^2 . \end{aligned}$$

Although this ridge problem is ideal, it can reveal some rules behind the two layer NN after we do the Taylor expansion. We replace the complex feature map $\boldsymbol{\Phi}$ by Gaussian design $\boldsymbol{Z}$

Li Weihao

CGMT-Han Qiyang

Li Weihao

inference in HD logistic regression with separated data

Can be used to make slides from her talk.

Model setting: Response variables $Y_1, \dots, Y_n$ are realisations satifying

$$\mathbb{P}(Y_i = 1) = \frac{e^{x_i^T \beta^*}}{1 + e^{x_i^T \beta^*}}, \quad \mathbb{P}(Y_i = -1) = 1 - \mathbb{P}(Y_i = 1)$$

for some unknown $\beta^* \in \mathbb{R}^p$ and covariates $x_1, \dots, x_n \in \mathbb{R}^p$. Let $X^T = (x_1, \dots, x_n)$.

- They study the behaviour of the OLS estimator
$$\hat{\beta}^0 = \left(X^T X\right)^{-1} X^T Y$$
assuming that Y consists of realisations from a logistic regression model. Can double descent hold for logistic regression with square loss
- The OLS estimator $\hat{\beta}^0$ converges to
$$\beta^0 = \left(X^T X\right)^{-1} X^T \tanh\left(X\beta^*/2\right).$$
- Write
$$\tanh\left(X\beta^*/2\right) = cX\beta^* + u + \Delta$$
where $u \in \mathrm{Col-Sp}(X)^{\perp}, P_M = M\left(M^T M\right)^{-1} M^T$ for any matrix M and
$$cX\beta^* = P_{X\beta^*} \tanh\left(X\beta^*/2\right), \quad \Delta = \left(P_X - P_{X\beta^*}\right) \tanh\left(X\beta^*/2\right).$$
Then, letting $\delta = \left(X^T X\right)^{-1} X^T \Delta$,
$$\beta^0 = c\beta^* + \delta.$$
$$\hat{\beta}^0 \to \beta^0 = c\beta^* + \delta.$$
- Main result is given by: Let $\hat{\eta}$ be a consistent estimator of $\eta = X\beta^* \neq 0$ satisfying(to make $\hat{\eta}$ consistent, we need to add some sparsity)
$$\max\left\{\frac{\|\hat{\eta} - \eta\|_2}{\|\eta\|_2}, \frac{\|\hat{\eta} - \eta\|_2}{\sqrt{n}}\right\} \xrightarrow{p} 0$$
as $p, n \to \infty$ with $p < n$. Define
$$\hat{c} = \frac{\hat{\eta}^T \tanh(\hat{\eta}/2)}{\|\hat{\eta}\|_2^2}, \quad \hat{\delta} = \left(X^T X\right)^{-1} X^T \hat{P} \tanh(\hat{\eta}/2),$$
where $\hat{P} = P_X - P_{\hat{\eta}}$. Finally, let
$$\tilde{\beta}^* = \hat{c}^{-1}\left(\hat{\beta}^0 - \hat{\delta}\right).$$
some theorem:
$$\frac{\alpha^T\left(\tilde{\beta}^* - \beta^*\right)}{c^{-1}\left\|\alpha^T \left(X^T X\right)^{-1} X^T \Gamma^{1/2}\right\|_2} \xrightarrow{d} N(0,1),$$
$$p^{-1/2}\left\|\tilde{\beta}^* - \beta^*\right\|_2 = o_P(1)$$

- **Interesting relationship with MLE**
  When the MLE $\hat{\beta}^*$ exists, correcting the least-squares estimator using $\hat{\eta} = X\hat{\beta}^*$ recovers the original MLE.

  In other words,
  $$\hat{\beta}^* = \frac{\hat{\beta}^0 - \hat{\delta}}{\hat{c}}$$
  where
  $$\hat{c} = \frac{(\hat{\eta})^T \tanh(\hat{\eta}/2)}{\|\hat{\eta}\|_2^2}, \quad \hat{\delta} = \left(X^T X\right)^{-1} X^T \hat{P} \tanh(\hat{\eta}/2),$$
  where $\hat{P} = P_X - P_{\hat{\eta}}$.
- Can be used in our paper: compare signal strength estimation, FDP

References

[AUDIBERT and TSYBAKOV, 2007] AUDIBERT, J.-Y. and TSYBAKOV, A. B. (2007). Fast learning rates for plug-in classifiers. *Annals of statistics*, 35(2):608–633.

[Cai et al., 2020] Cai, Z., Li, R., and Zhu, L. (2020). Online sufficient dimension reduction through sliced inverse regression. *The Journal of Machine Learning Research*, 21(1):321–345.

[Chen et al., 2020] Chen, X., Lee, J. D., Tong, X. T., and Zhang, Y. (2020). Statistical inference for model parameters in stochastic gradient descent. *Annals of Statistics*, 48(1):251–273.

[Koltchinskii, 2006] Koltchinskii, V. (2006). Local rademacher complexities and oracle inequalities in risk minimization. *The Annals of Statistics*, pages 2593–2656.

[Polyak and Juditsky, 1992] Polyak, B. T. and Juditsky, A. B. (1992). Acceleration of stochastic approximation by averaging. *SIAM journal on control and optimization*, 30(4):838–855.

[Ruppert, 1988] Ruppert, D. (1988). Efficient estimations from a slowly convergent robbins-monro process. Technical report, Cornell University Operations Research and Industrial Engineering.

[TSYBAKOV and VAN DE GEER, 2005] TSYBAKOV, A. and VAN DE GEER, S. (2005). Square root penalty: Adaptation to the margin in classification and in edge estimation. *Annals of statistics*, 33(3):1203–1224.

[Tsybakov, 2004] Tsybakov, A. B. (2004). Optimal aggregation of classifiers in statistical learning. *The Annals of Statistics*, 32(1):135–166.

[Yang, 1999] Yang, Y. (1999). Minimax nonparametric classification. i. rates of convergence. *IEEE Transactions on Information Theory*, 45(7):2271–2284.