CMSC5741 Big Data Tech. & Apps.

Lectur arning

Prof. Michael R. Lyu

Computer Science & Engineering Dept.

The Chinese University of Hong Kong

A Motivating Example– Spam Filtering

Incoming Emails

Spam Filter

Not Spams

Spams

Traditional Method: Training

A batch of Emails

Feature extraction: X

Incoming Emails

Linear classifier: $W^T X$

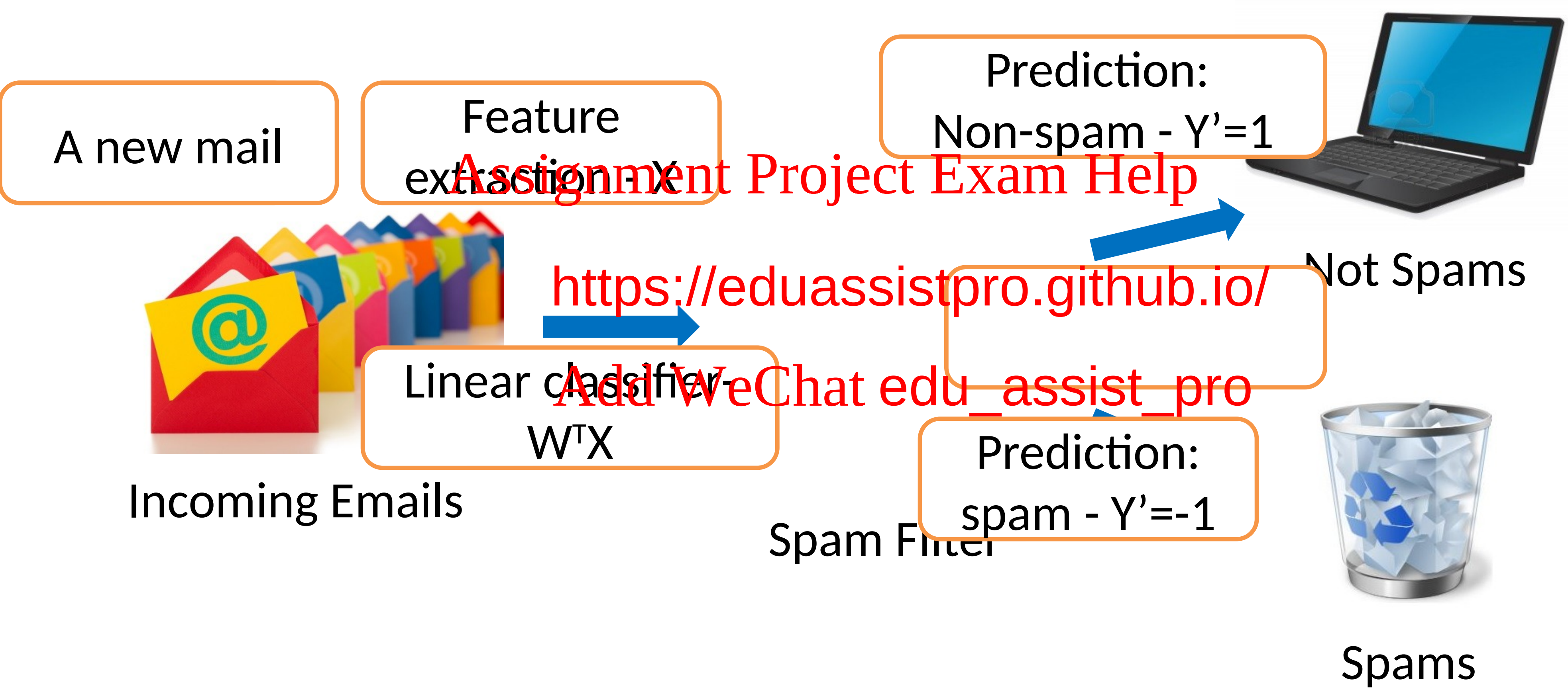
Spam Filter

Model selection

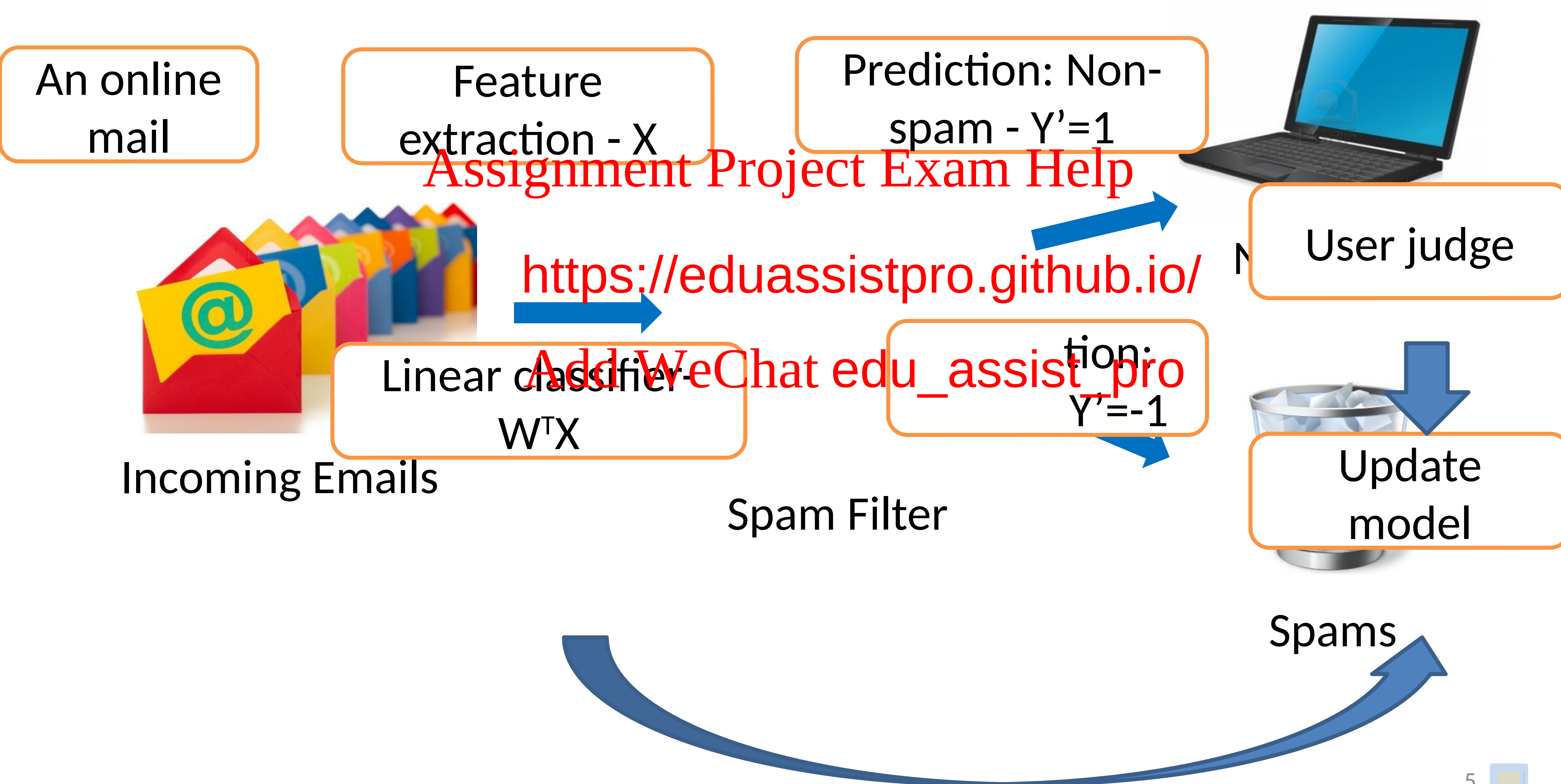
Not Spams

Spams

Traditional Method: Test



Online Protocol



Outline

- Introduction
  - Learning paradigms
  - Online learning and its applications
- Online learnin
  - Perceptron
  - Online non-sparse learning
  - Online sparse learning
  - Online unsupervised learning
- Conclusion

Outline

- Introduction
  - Learning paradigms
  - Online learning and its applications
- Online learnin
  - Perceptron
  - Online non-sparse learning
  - Online sparse learning
  - Online unsupervised learning
- Conclusion

Outline

- Introduction
  - Learning paradigms
  - Online learning and its applications
- Online learnin
  - Perceptron
  - Online non-sparse learning
  - Online sparse learning
  - Online unsupervised learning
- Conclusion

Learning Paradigms Overview

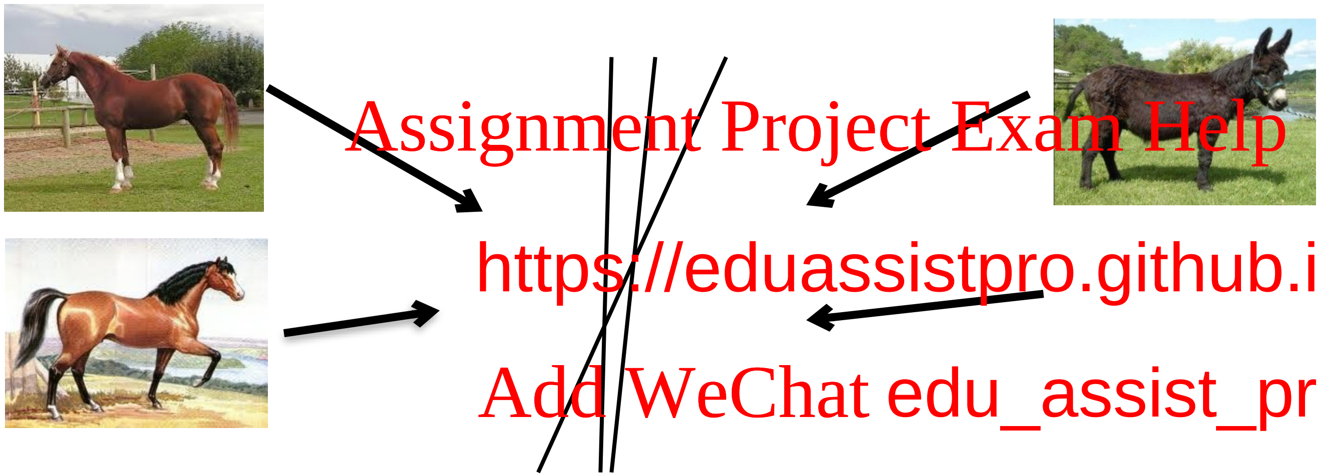
- Learning paradigms
  - **Supervised learning**

Learning Paradigms Overview

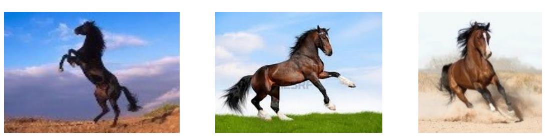
- Learning paradigms
  - **Supervised learning**
  - Semisupervised
  - Transductive learning
  - **Unsupervised learning**
  - Universum learning
  - Transfer learning

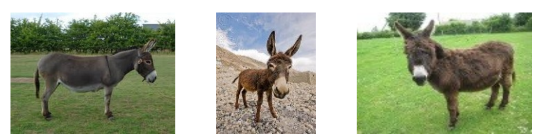
Supervised Learning

- Train on labeled data



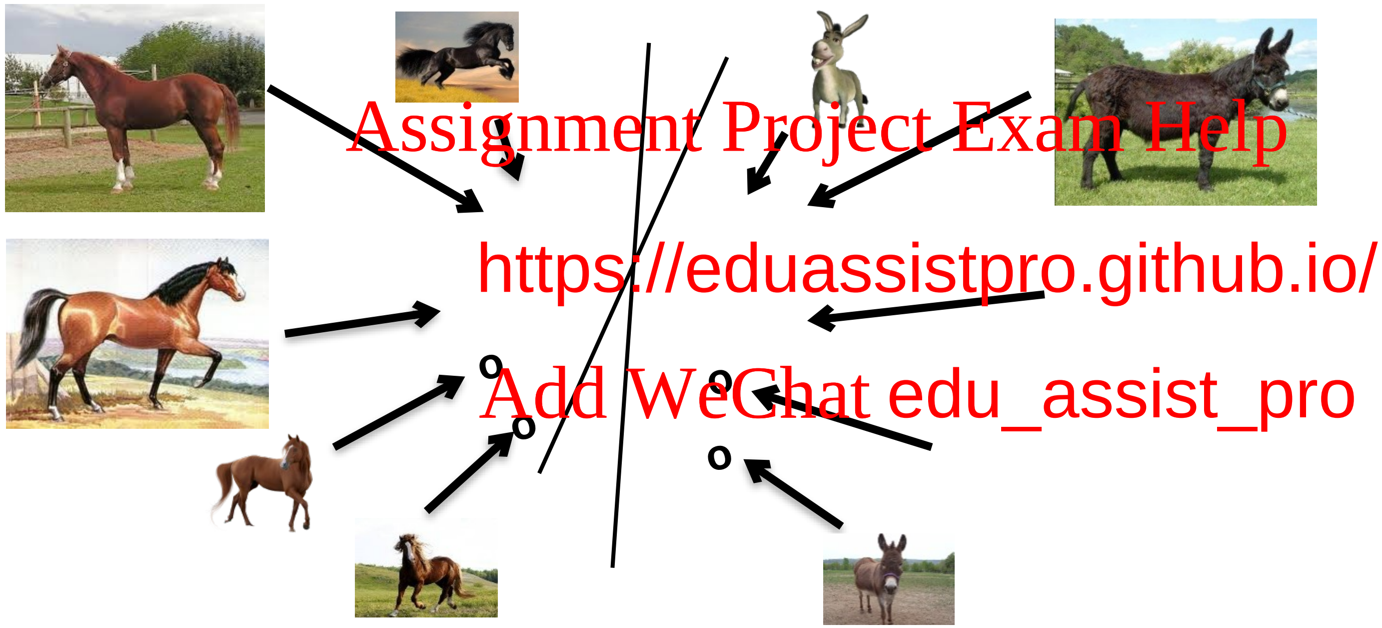
- Test on test data

 ?

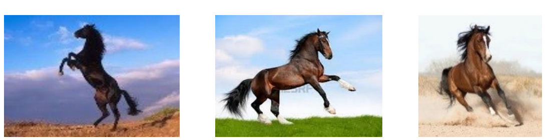
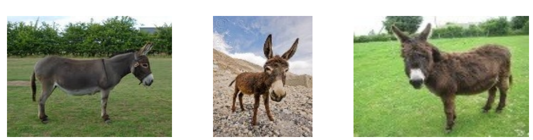
 ?

Semisupervised Learning

- Train on labeled and unlabeled data



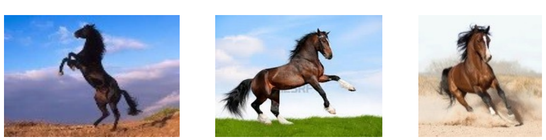
- Test on test data

 ?  ?

Transductive Learning

- Train on labeled and test data

- Test on test data



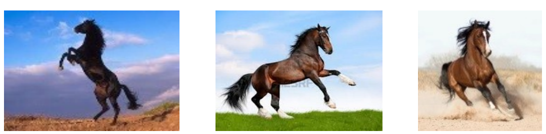
?

?

Unsupervised Learning

- Train on unlabeled data

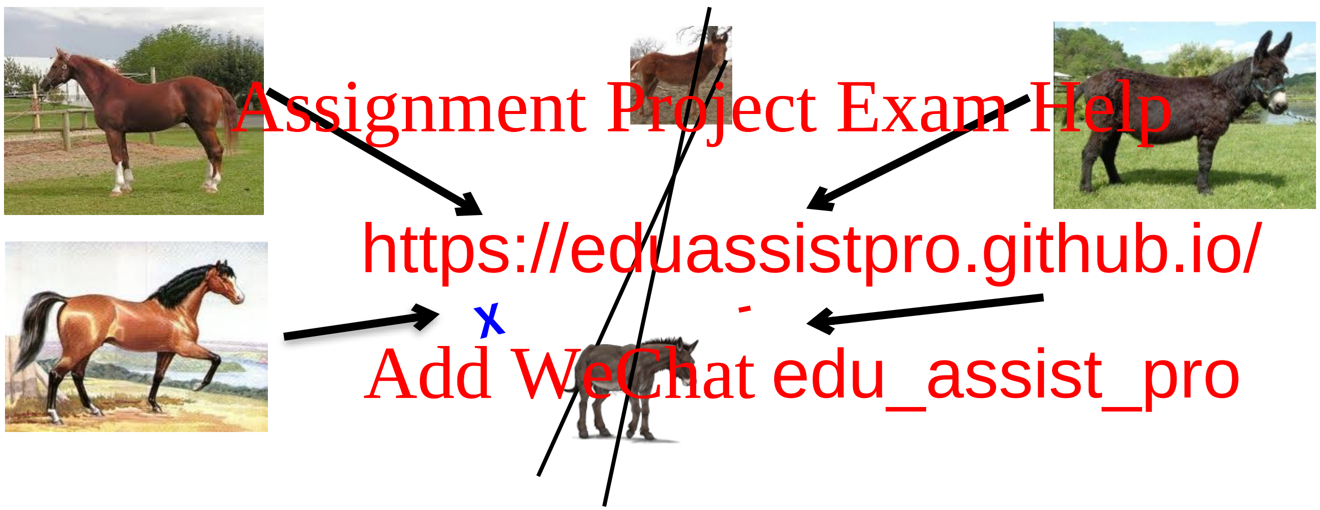
- Test on test data (Test reconstruction error)

 ?

 ?

Universum Learning

- Train on labeled and universum data



- Test on test data

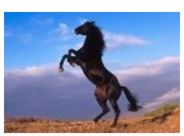
Transfer Learning

- Train on labeled from source and target domains





- Test on test data




?




?

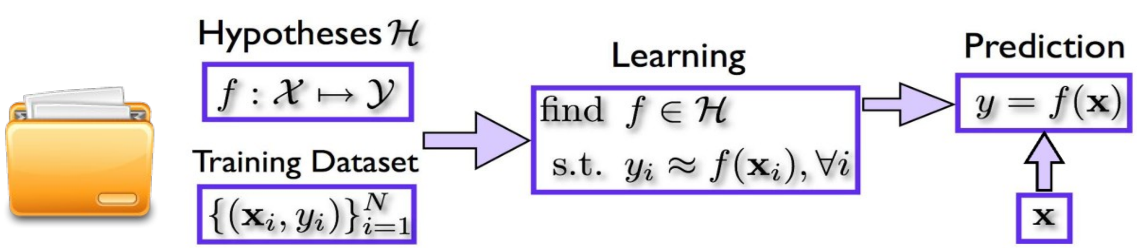
Outline

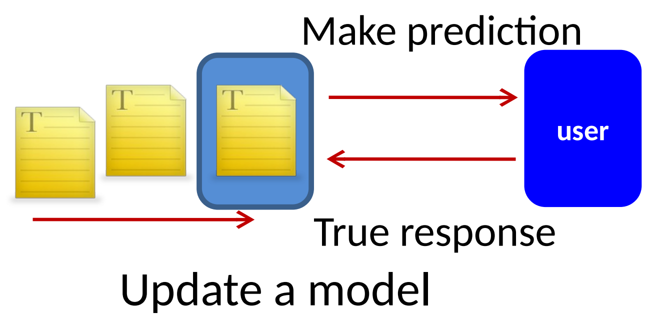
- Introduction
  - Learning paradigms
  - Online learning and its applications
- Online learnin
  - Perceptron
  - Online non-sparse learning
  - Online sparse learning
  - Online unsupervised learning
- Conclusion

What is Online Learning?

- Batch/Offline learning
  - Observe a **batch** of training data $\{(\mathbf{x}_i, y_i)\}_{i=1}^N$
  - Learn a model fro
  - Predict new sampl

- Online learning
  - Observe a **sequence** of data $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_t, y_t)$
  - a model **incrementally** as
  - es come
  - e sequence of online
  - ons accurately

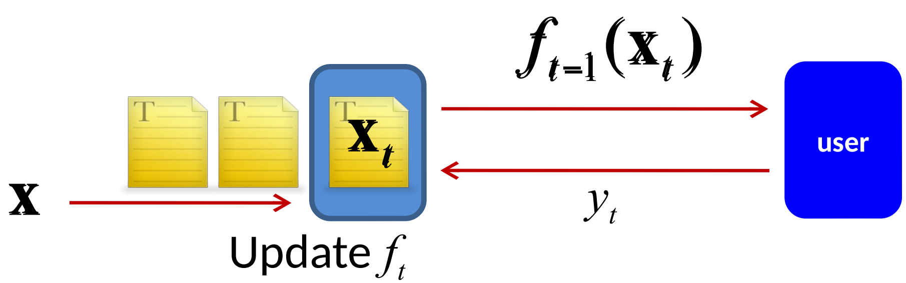




Online learning is the process of answering a sequence of questions given (maybe partial) knowledge of the correct answers to previous questions and possibly additional available information. [Shal11]

Online Prediction Algorithm

- An initial prediction rule $f_0(\cdot)$
- For t = 1, 2, …
  - We observe $\mathbf{x}$ n $f_{t-1}(\mathbf{x}_t)$
  - We observe t d then compute a loss $l(f_{t-1}(\mathbf{x}_t), y_t)$
  - The online algorithm updates the prediction rule using the new example and construct $f_t(\mathbf{x})$

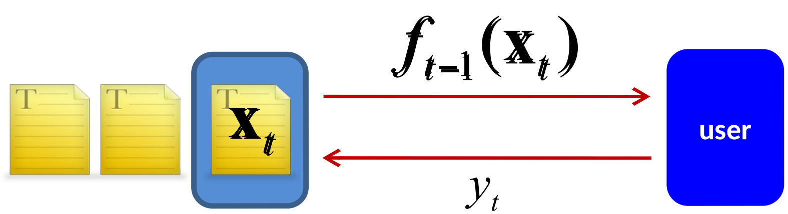


Online Prediction Algorithm

- The total error of the method is

$$\sum_{t=1}^{T} l(f_{t-1}(\mathbf{x}_t), y_t)$$

- Goal: this erro ble
- Predict unknown future o time: similar to generalization error



Regret Analysis

- $f_*(\cdot)$: optimal prediction function from a class H, e.g., the class of linear classifiers

$f_*(\cdot)$

with minimum ll examples

- Regret for the online lear ithm

$$\textbf{regret} = \frac{1}{T}\sum_{t=1}^{T}\left[l(f_{t-1}(\mathbf{x}_t), y_t) - l(f_*(\mathbf{x}_t), y_t)\right]$$

We want regret as small as possible

Why Low Regret?

- Regret for the online learning algorithm

$$\text{regret} = \frac{1}{T}\sum_{t=1}^{T}\left[l(f_{t-1}(\mathbf{x}_t), y_t) - l(f_*(\mathbf{x}_t), y_t)\right]$$

- Advantages
  - We do not lose much from g future events
  - We can perform almost as well as someone who observes the entire sequence and picks the best prediction strategy in hindsight
  - We can also compete with changing environment

Advantages of Online Learning

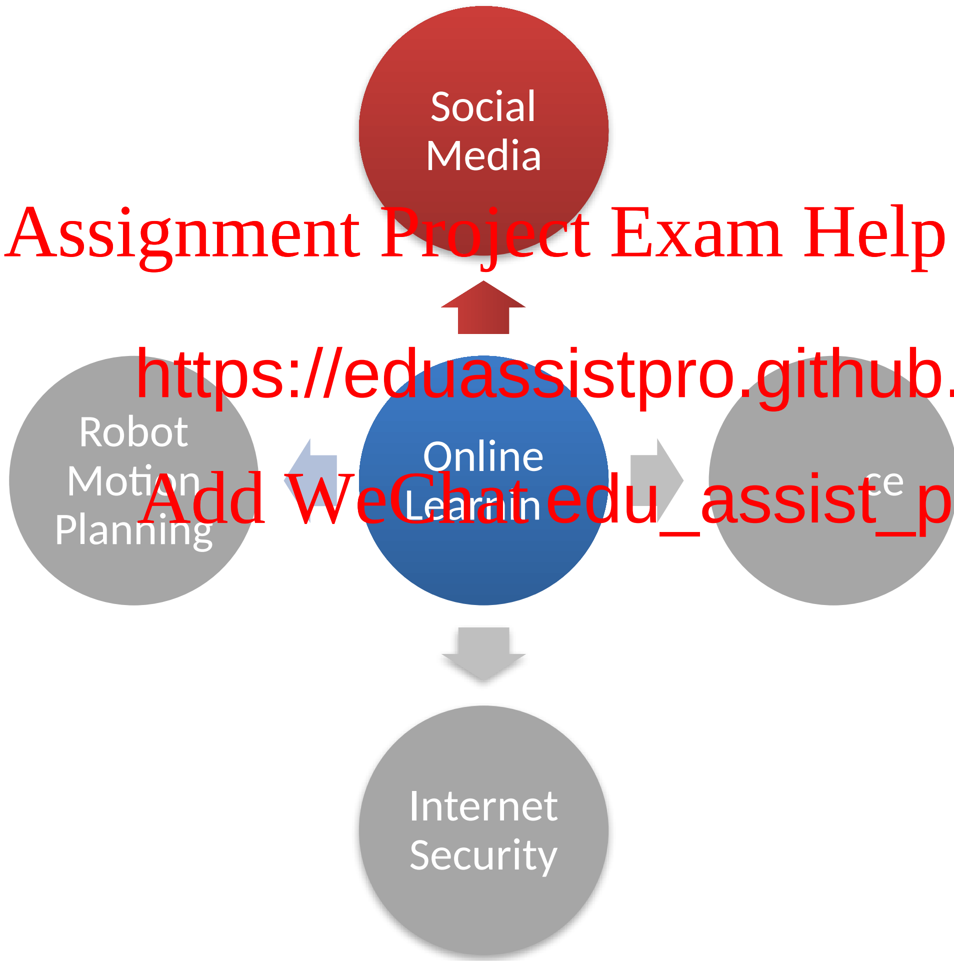
- Meet many applications for data arriving sequentially while predictions are required on-the-fly
  - Avoid re-training when adding new data

- Applicable in ad tive environment

- Strong adaptability to changi ment

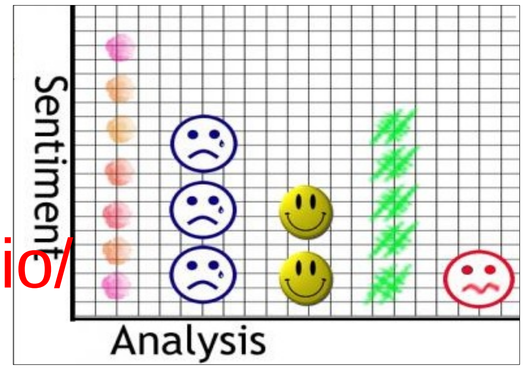
- High efficiency and excellent scalability
- Simple to understand and easy to implement
- Easy to be parallelized
- Theoretical guarantees

Where to Apply Online Learning?

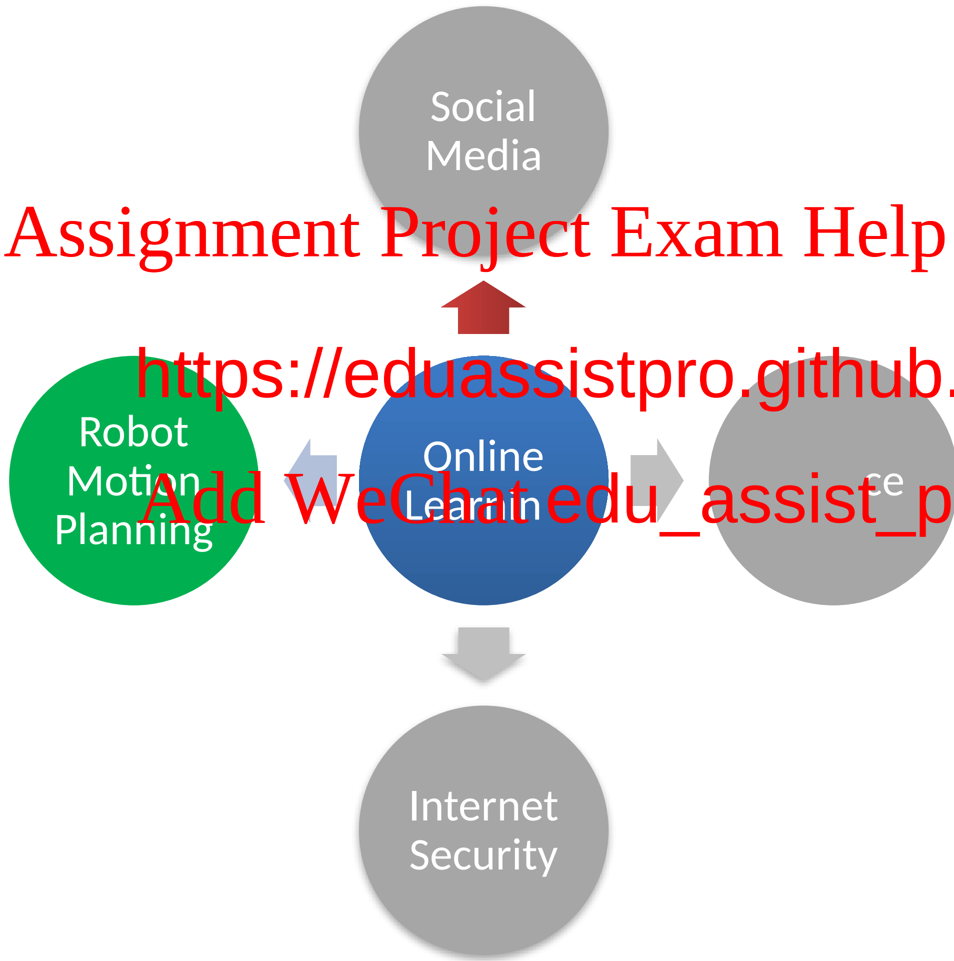


Online Learning for Social Media

- Recommendation, sentiment/emotion analysis

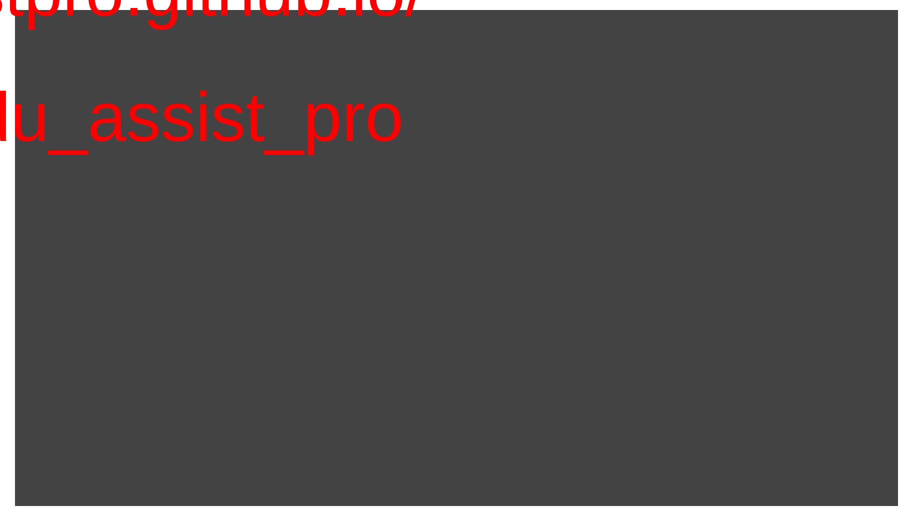


Where to Apply Online Learning?



Online Learning for Robot Motion Planning

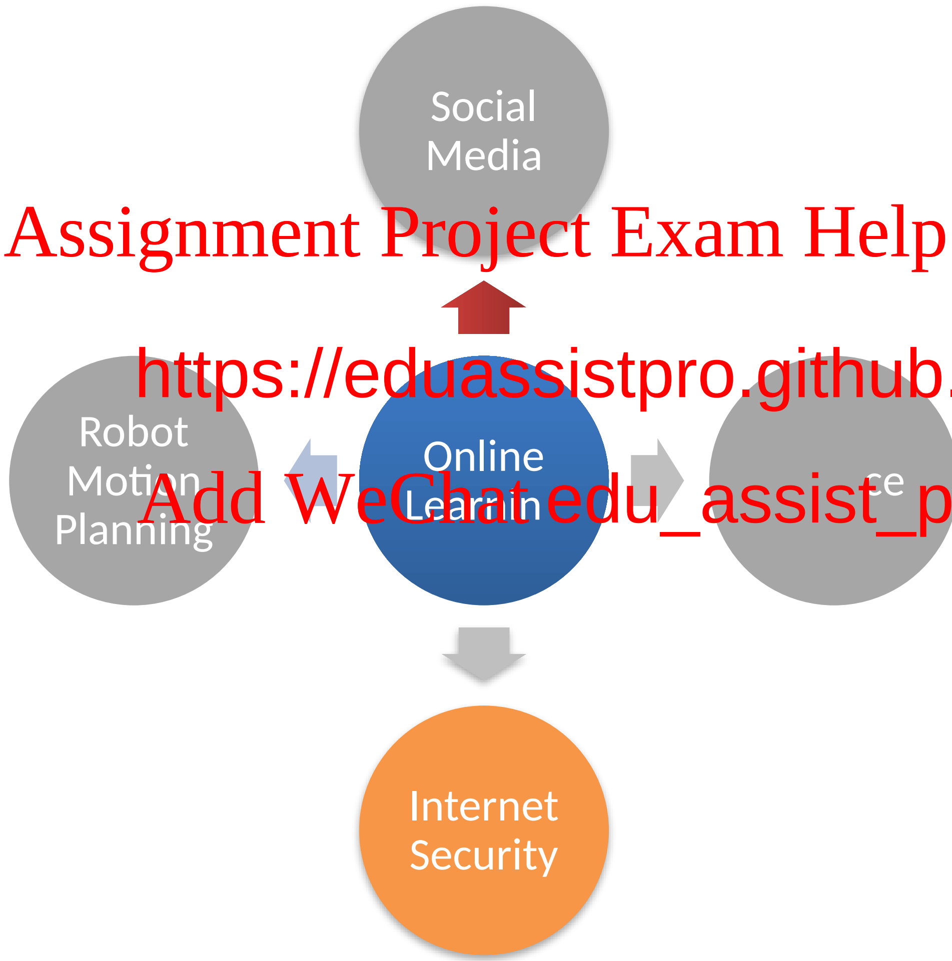
- Tasks
  - Exploring an unknown terrain
  - Finding a destination



Rock-Paper-Scissors: You vs. the Computer

Robot Dog

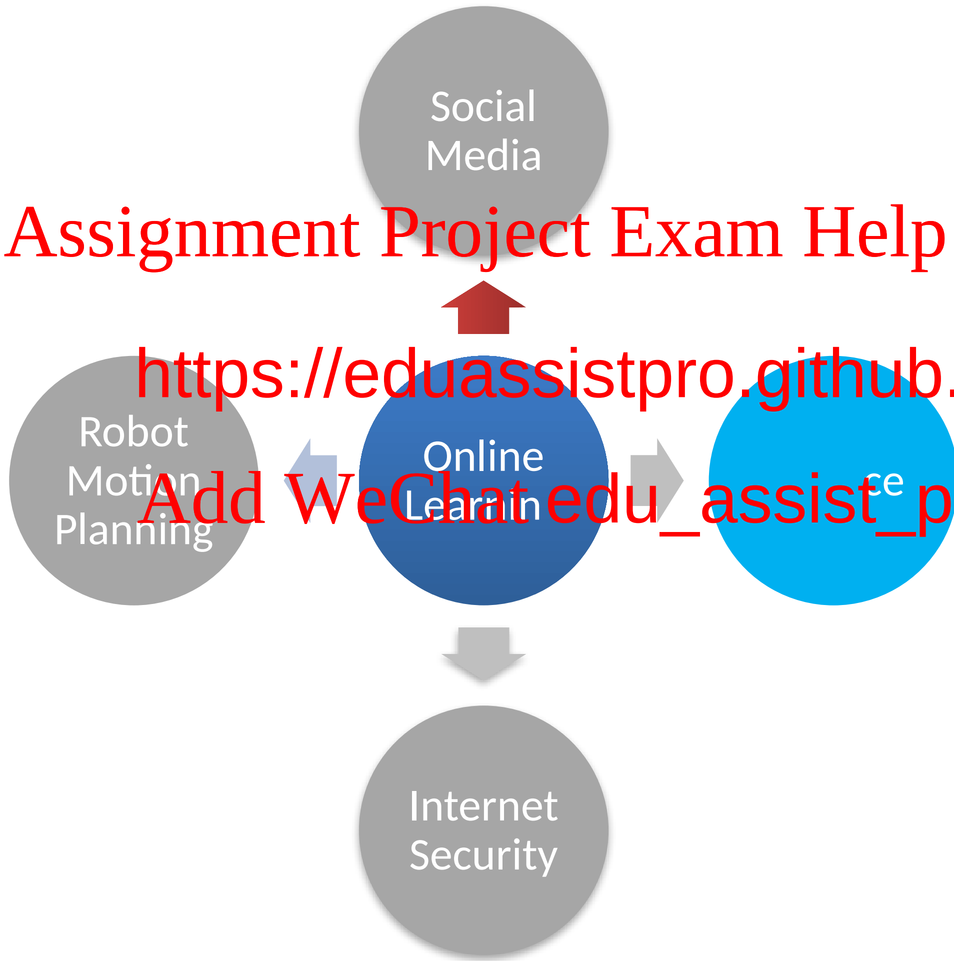
Where to Apply Online Learning?



Online Learning for Internet Security

- Electronic business sectors
  - **Spam email** filtering
  - **Fraud** credit tion
  - Network **intr** , etc.

Where to Apply Online Learning?



Online Learning for Financial Decision

- Financial decision
  - Online portfolio selection
  - Sequential in

Outline

- Introduction
  - Learning paradigms
  - Online learning and its applications
- Online learning
  - Perceptron
  - Online non-sparse learning
  - Online sparse learning
  - Online unsupervised learning
- Conclusion

Outline

- Introduction
  - Learning paradigms
  - Online learning and its applications
- Online learnin
  - Perceptron
  - Online non-sparse learning
  - Online sparse learning
  - Online unsupervised learning
- Conclusion

Perceptron Algorithm (F. Rosenblatt, 1958)

- One of the oldest machine learning algorithm
- Online algorithm for learning a linear threshold function with

n

Perceptron Algorithm (F. Rosenblatt, 1958)

- Goal: find a linear classifier with small error

If no error, keeping the same;
otherwise, update.

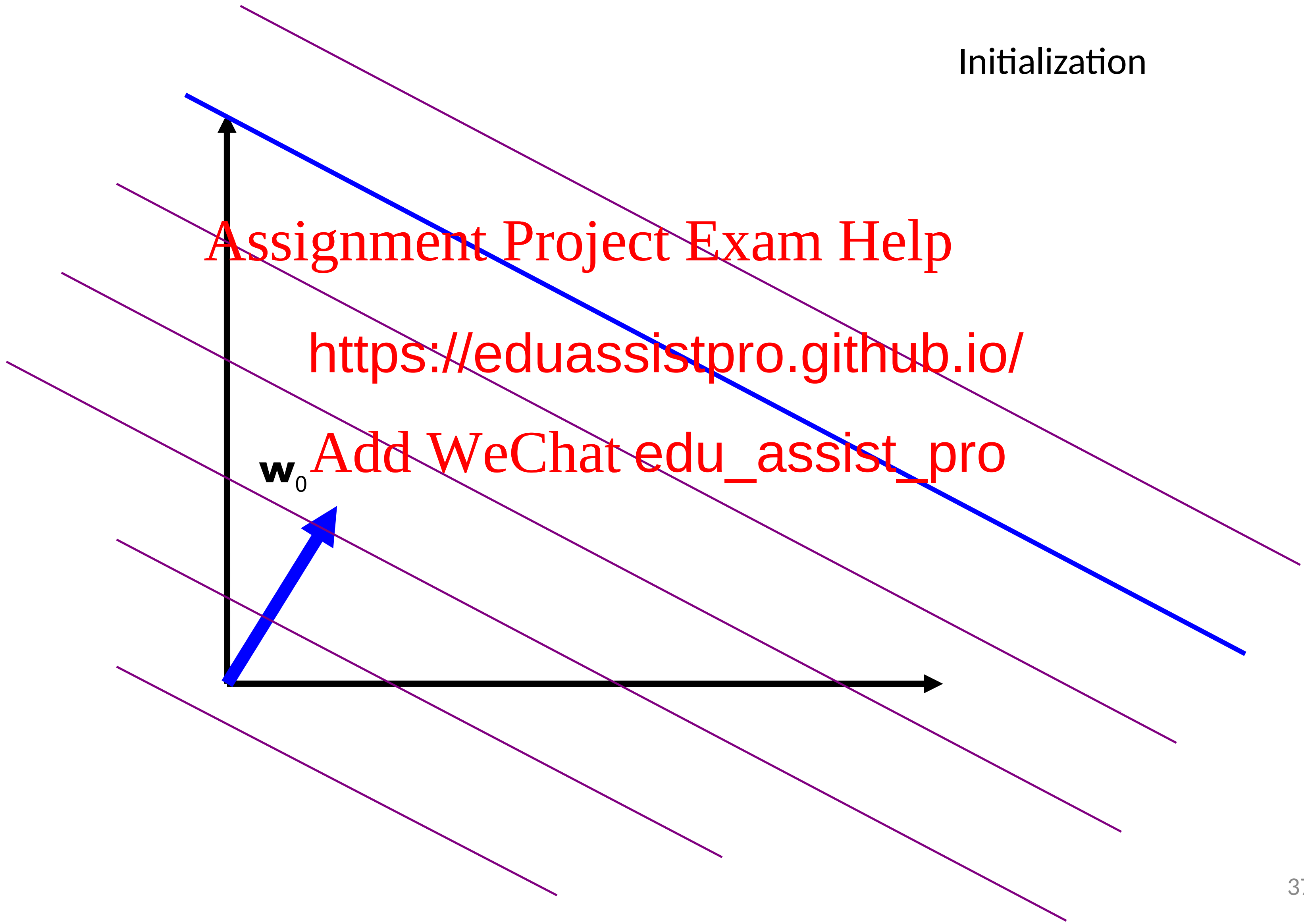
Intuition Explanation

- Want positive margin:

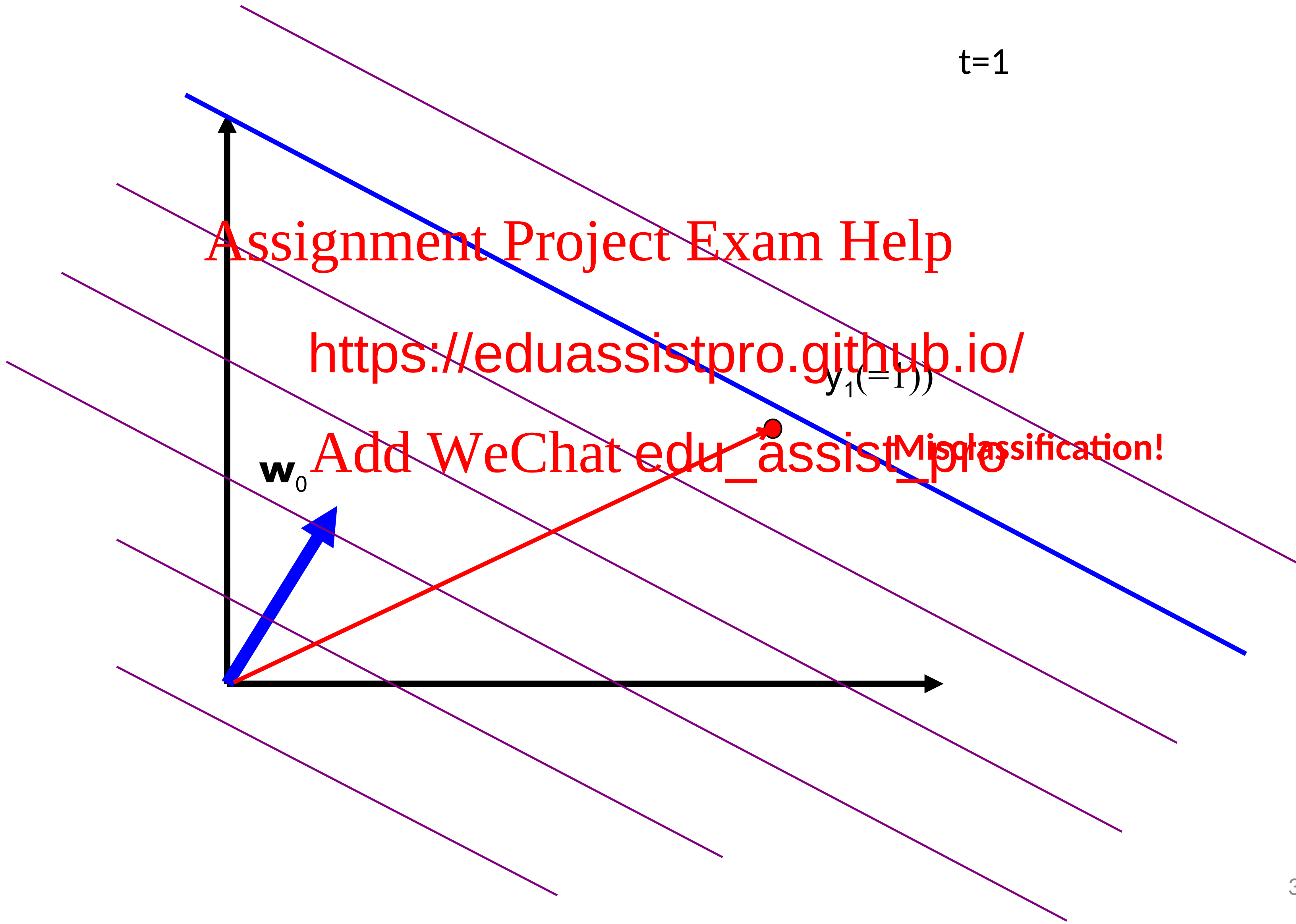
- Effect of Perc argin:

- So margin increases

Geometric View

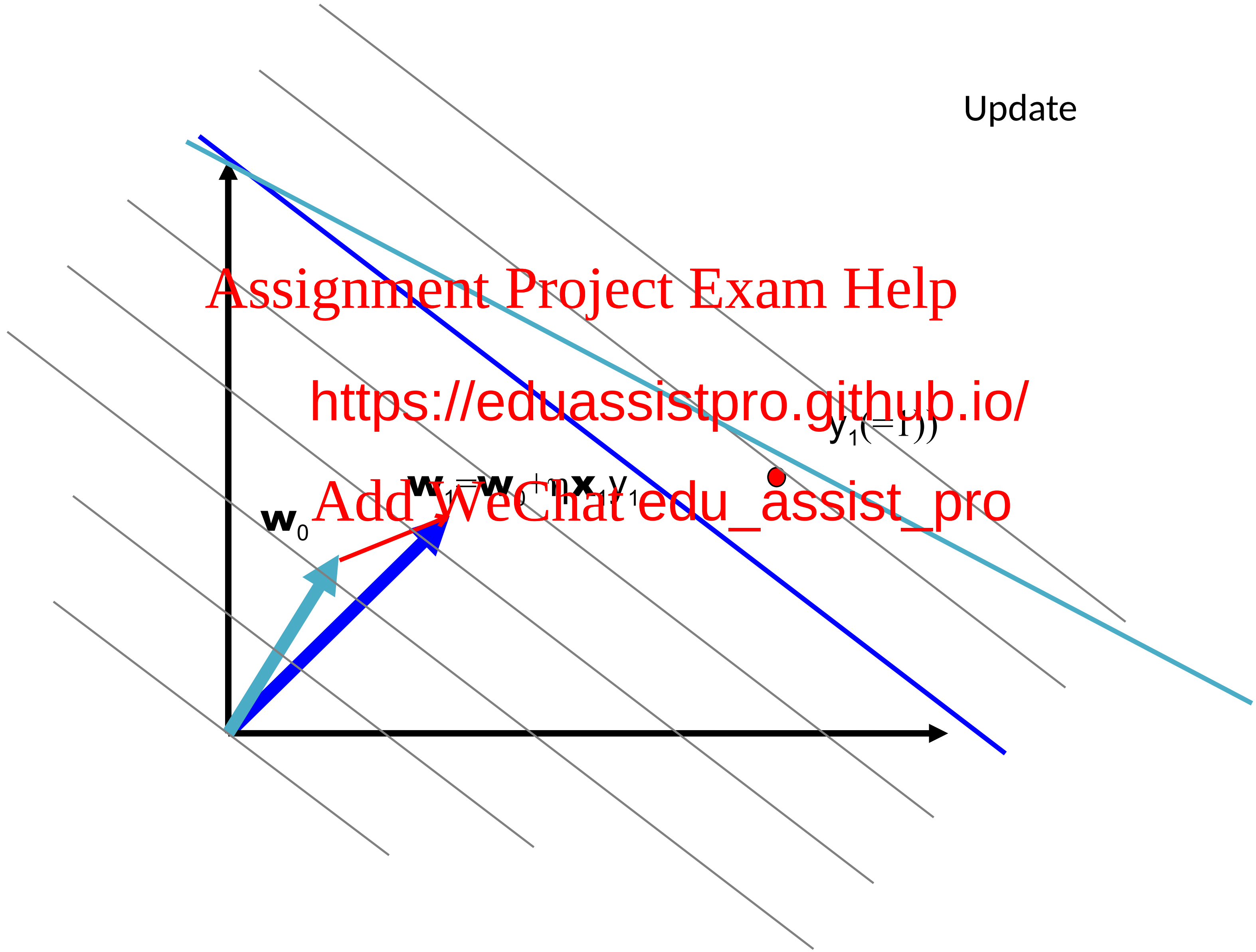


Geometric View

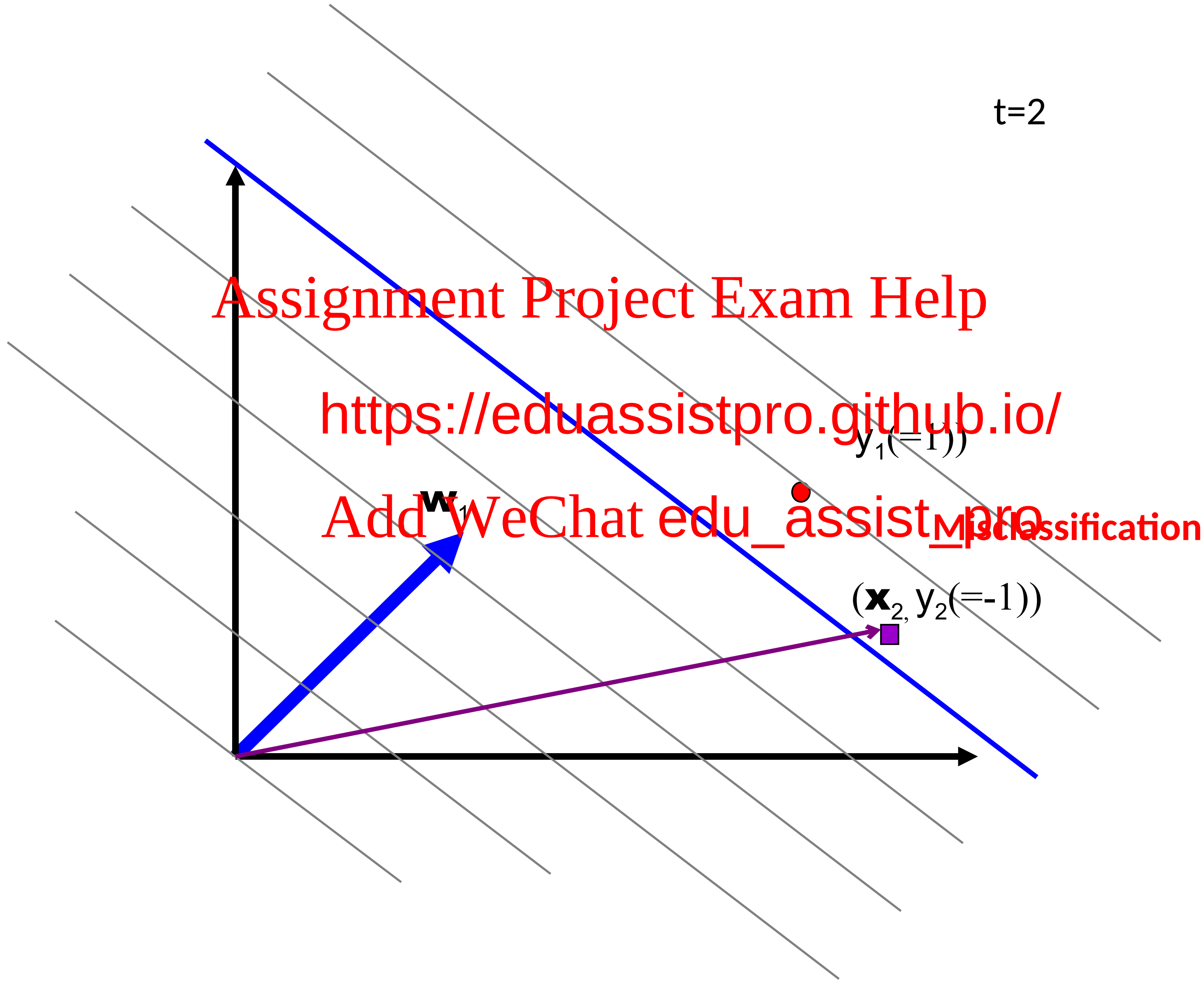


Geometric View

Update



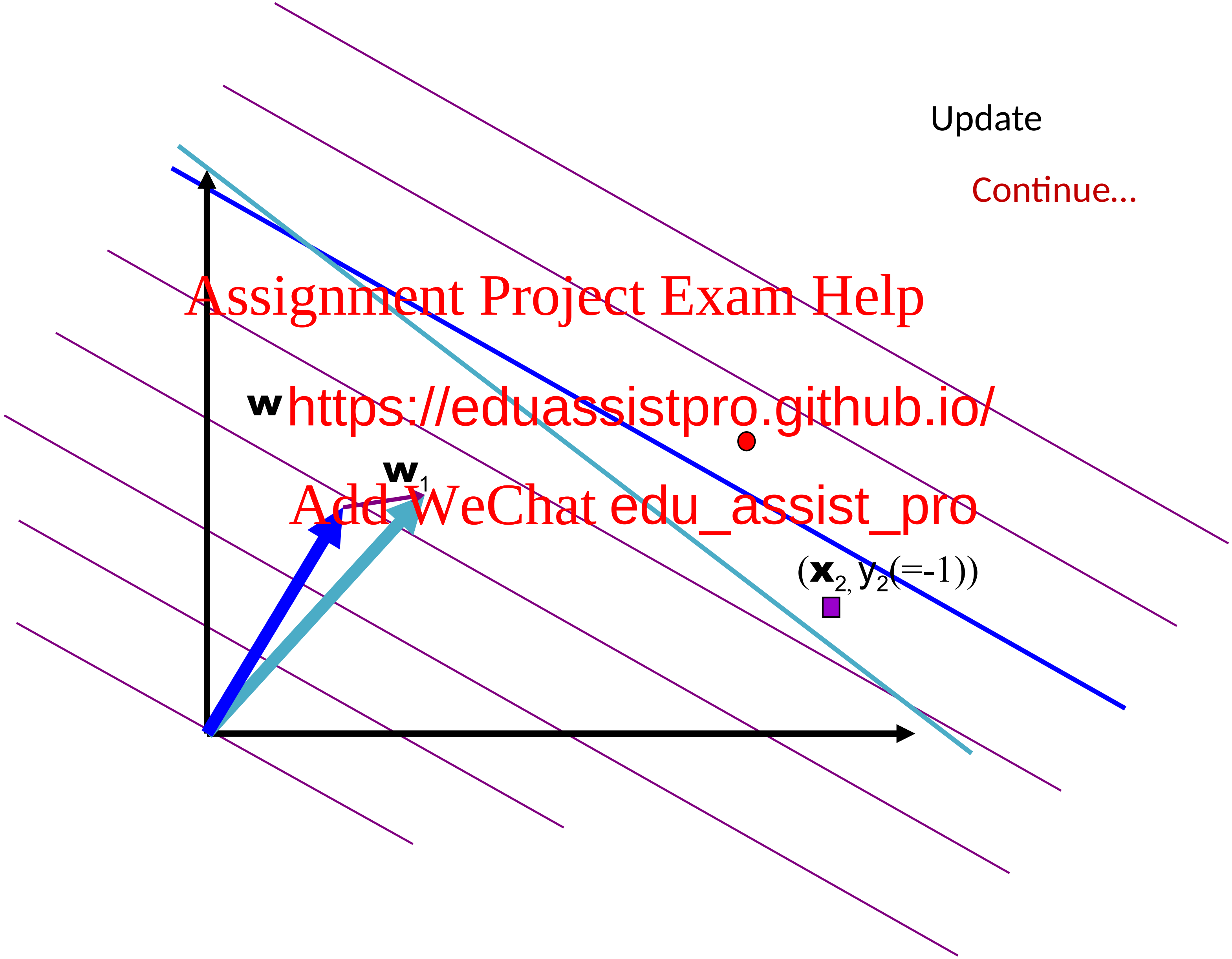
Geometric View



Geometric View

Update

Continue...



In-class Practice

- Go to practice

Perceptron Mistake Bound

- Consider w_* separate the data: $\mathbf{w}_*^T \mathbf{x}_i y_i > 0$
- Define margin

$$\gamma \quad \frac{\min_i |\mathbf{w}_*^T \mathbf{x}_i|}{\|\mathbf{w}_*\| \; \max_i \|\mathbf{x}_i\|}$$

The larger, the more confidence

rm of x: the larger, the mistake bound

- The number of mistakes p n makes is at most γ^{-2}

Proof of Perceptron Mistake Bound [Novikoff, 1963]

Proof: Let $\mathbf{v}_k$ be the hypothesis before the k-th mistake. Assume that the k-th mistake occurs on the input example $(\mathbf{x}_i, y_i)$

$$\gamma = \frac{\min_i \left|\mathbf{w}_*^T \mathbf{x}_i\right|}{\left\|\mathbf{w}_*\right\|_2 \sup_i \left\|\mathbf{x}_i\right\|_2}$$

First,

$$\begin{aligned}
\|\mathbf{v}_{k+1}\|^2 &= \|\mathbf{v}_k + y_i \mathbf{x}_i\|^2 \\
&= \|\mathbf{v}_k\|^2 + 2y_i(\mathbf{v}_k^T \mathbf{x}_i) \\
&\quad + \|\mathbf{x}_i\|^2 \\
&\leq \|\mathbf{v}_k\|^2 + R^2 \\
&\leq kR^2 (R := \sup_i \|\mathbf{x}\|_2)
\end{aligned}$$

$$\mathbf{v}_{k+1}^T \mathbf{u} \geq k\gamma R.$$

Hence, $\sqrt{k}R \geq \|\mathbf{v}_{k+1}\| \geq \mathbf{v}_{k+1}^T \mathbf{u} \geq k\gamma R$

$$k \leq \gamma^{-2}$$

Outline

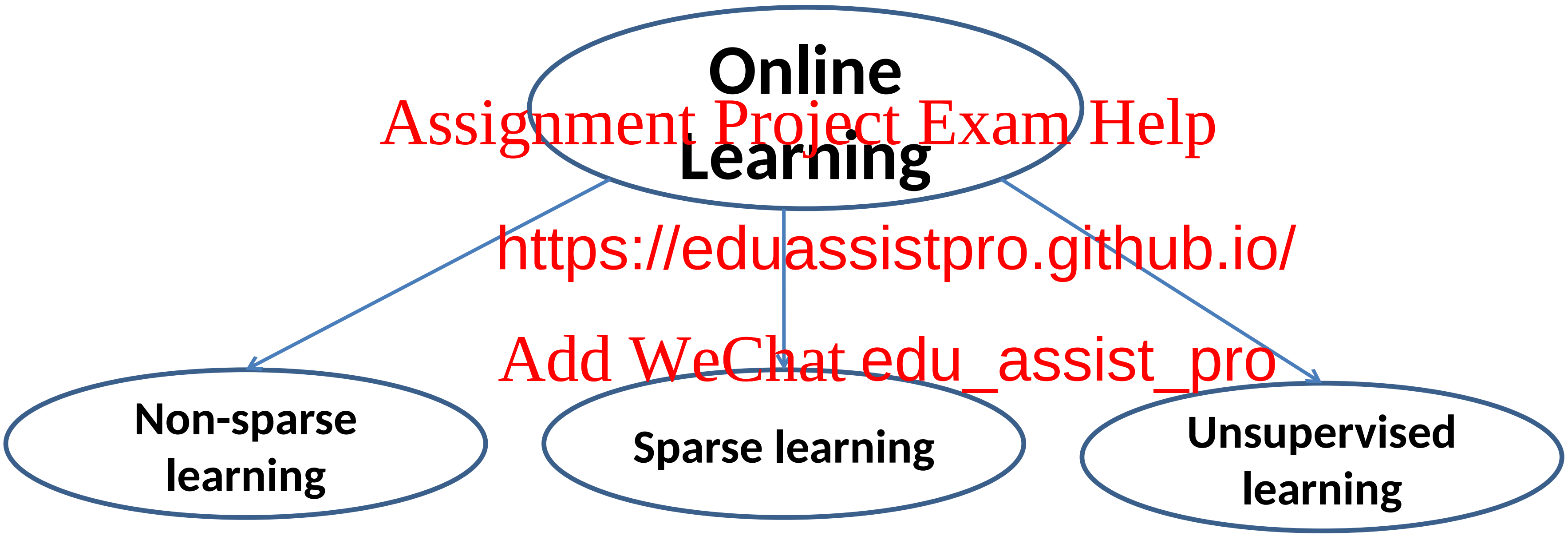
- Introduction
  - Learning paradigms

  - Online learning and its applications
- Online learnin

  - Perceptron

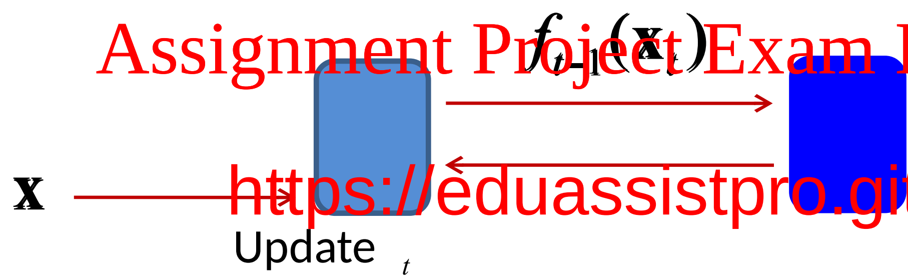
  - Online non-sparse learning
  - Online sparse learning
  - Online unsupervised learning
- Conclusion

Overview

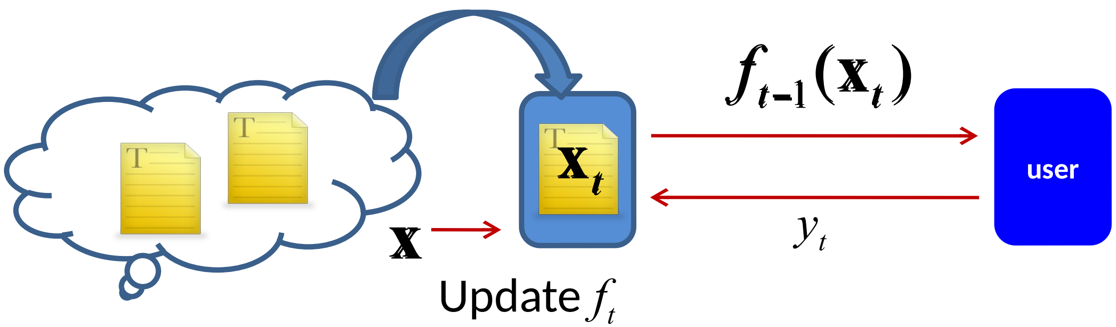

Online
Learning
Non-sparse
learning
Sparse learning
Unsupervised
learning

Online/Stochastic Gradient Descent

- Online gradient descent



- Stochastic gradient desce



Outline

- Introduction
  - Learning paradigms
  - Online learning and its applications

- Online learnin

  - Perceptron

  - Online non-sparse learning
  - Online sparse learning
  - Online unsupervised learning
- Conclusion

Online Non-Sparse Learning

- Decision function can be linear and non-linear as

- **First order** learning

  - **Online gradient de**
  - **Passive aggressive learning** (Cramm)

  - Others (including but not limited)
    - ALMA: A New Approximate Maximal Margin Classification Algorithm (Gentile, 2001)
    - ROMMA: Relaxed Online Maximum Margin Algorithm (Li and Long, 2002)
    - MIRA: Margin Infused Relaxed Algorithm (Crammer and Singer, 2003)
    - DUOL: A Double Updating Approach for Online Learning (Zhao et al., 2009)

Online Gradient Descent (OGD)
(Zinkevich, 2003)

- Online convex optimization
  - Consider a convex objective function

where

et

  - Update by Online Gradient Descent (OGD) or Stochastic Gradient Descent (SGD)

$$\mathbf{w}_{t+1} \leftarrow \prod_S \left(\boxed{\mathbf{w}_t - \eta \nabla f(\mathbf{w}_t)} \right)$$

projection

gradient descent

where η is a learning rate

Provide a framework to prove regret bound for online convex optimization

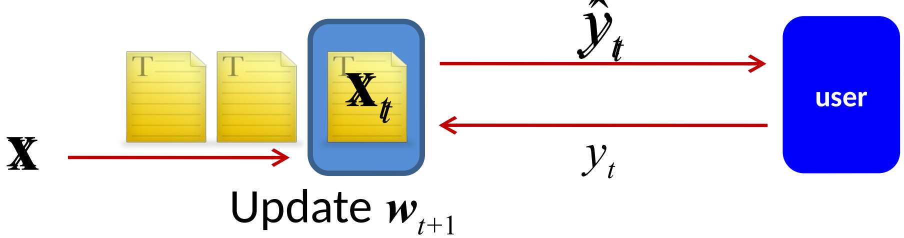
Online Gradient Descent (OGD) (Zinkevich, 2003)

- For $t = 1, 2, \ldots$
  - An unlabeled sample $\mathbf{x}_t$ arrives
  - Make a predi g weights
  - Observe the true class labe 1,+1}
  - Update the weights by

$$\mathbf{w}_{t+1} \leftarrow \prod_{S}(\mathbf{w}_t - \eta \nabla f(\mathbf{w}_t))$$

where η is a learning rate

regret bound is established.

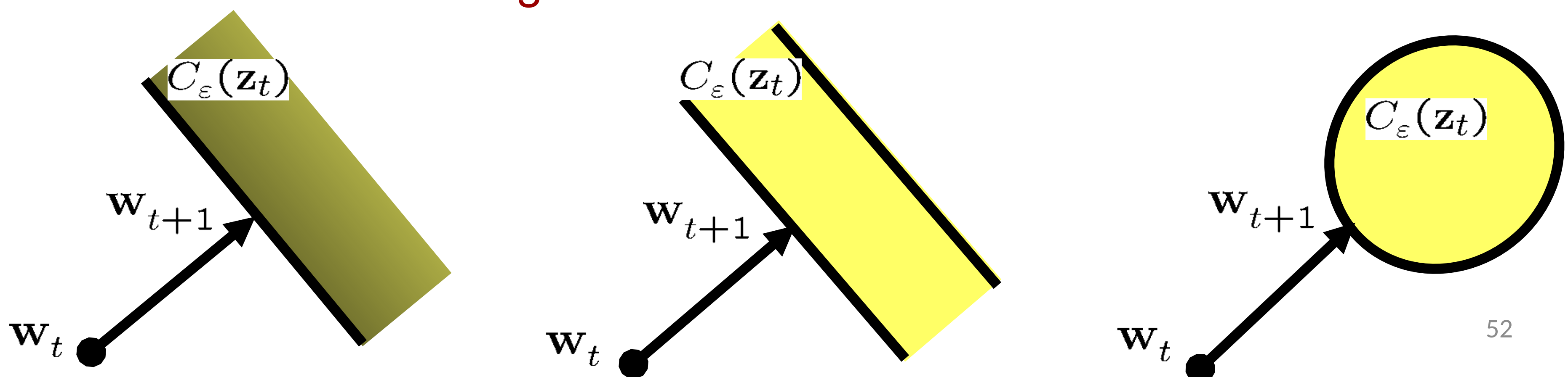


Passive-Aggressive Online Learning (Crammer et al., 2006)

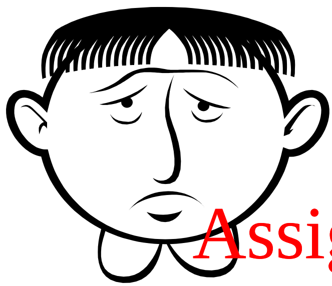
- Each example defines a set of consistent hypotheses: $C_\varepsilon(\mathbf{z}_t) = \{\mathbf{w} \mid \delta(\mathbf{w};\mathbf{z}_t) \leq \varepsilon\}$

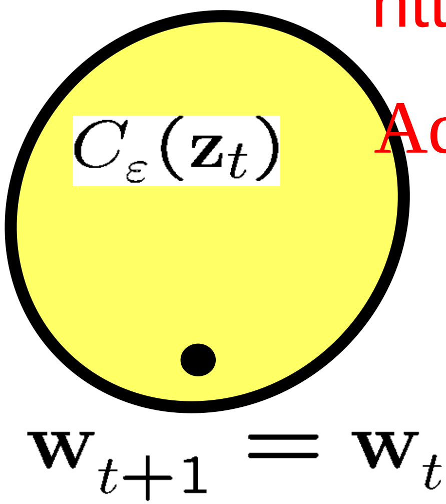
- The new vect the projection of $\mathbf{w}_t$ onto

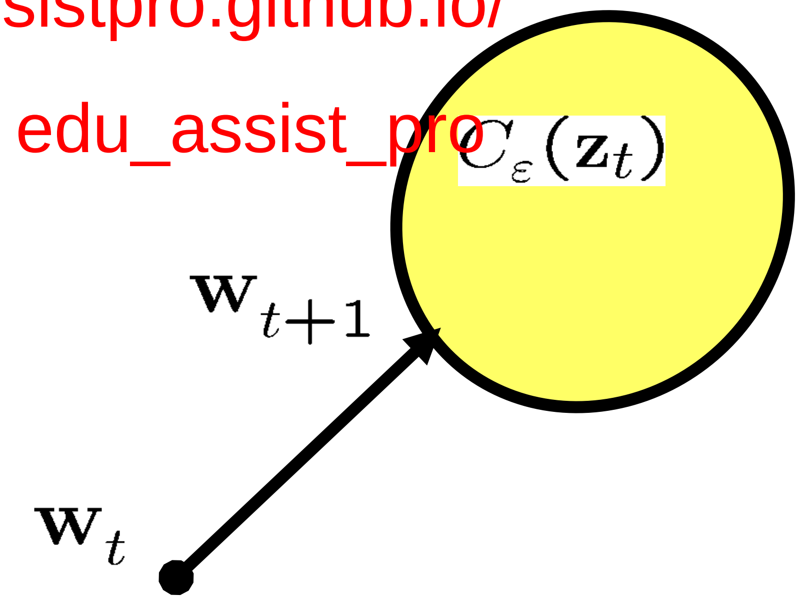
Classification Regression Uniclass



Passive-Aggressive







Passive Aggressive Online Learning
(Crammer et al., 2006)

- PA (Binary classification)

$$\mathbf{w}_{t+1} = \operatorname*{argmin}_{\mathbf{w}\in\mathbb{R}^n} \frac{1}{2}\|\mathbf{w}-\mathbf{w}_t\|^2$$

- Closed-form solution

$$\mathbf{w}_{t+1} = \mathbf{w}_t + \tau_t y_t \mathbf{x}_t$$

- PA-I (C-SVM)

$$\mathbf{w}_{t+1} = \operatorname*{argmin}_{\mathbf{w}\in\mathbb{R}^n} \frac{1}{2}\|\mathbf{w}-\mathbf{w}_t\|^2 + C\xi$$

$$\text{s.t.} \quad \ell(\mathbf{w};(\mathbf{x}_t,y_t)) \le \xi \ \text{ and } \ \xi \ge 0.$$

- PA-II (Relaxed C-SVM)

$$\mathbf{w}_{t+1} = \operatorname*{argmin}_{\mathbf{w}\in\mathbb{R}^n} \frac{1}{2}\|\mathbf{w}-\mathbf{w}_t\|^2 + C\xi^2$$

$$\text{s.t.} \quad \ell(\mathbf{w};(\mathbf{x}_t,y_t)) \le \xi.$$

Passive Aggressive Online Learning
(Crammer et al., 2006)

- Algorithm

- Objective

$$\mathbf{w}_{t+1} = \underset{\mathbf{w}\in\mathbb{R}^n}{\operatorname{argmin}} \frac{1}{2}\|\mathbf{w}-\mathbf{w}_t\|^2$$

$$\text{s.t.} \quad \ell(\mathbf{w};(\mathbf{x}_t,y_t)) = 0$$

- Closed-form solutions

$$\tau_t = \frac{\ell_t}{\|\mathbf{x}_t\|^2} \qquad \text{(PA)}$$

$$\tau_t = \min\left\{C\,,\ \frac{\ell_t}{\|\mathbf{x}_t\|^2}\right\} \qquad \text{(PA-I)}$$

$$\tau_t = \frac{\ell_t}{\|\mathbf{x}_t\|^2+\frac{1}{2C}} \qquad \text{(PA-II)}$$

Online Non-Sparse Learning

- **First order** methods
  - Learn a **linear** weight vector (first order) of model

- Pros and Cons
  - Simple and e

  - Efficient and scalable for sional data

  - Relatively slow convergence rate

Online Non-Sparse Learning

- **Second order** online learning methods
  - Update the weight vector **w** by maintaining and exploring both first-order and second-order information
- Some representative methods, but not limited
  - SOP: Second Order Perceptron (Cesa-Bianchi et al., 2005)
  - CW: Confidence Weighted l
  - AROW: Adaptive Regulariza)
  - IELLIP: Online Learning by Ellipsoid Method (Yang e
  - NHERD: Gaussian Herding (Crammer & Lee 2010)
  - NAROW: New variant of AROW algorithm (Orabona & Crammer 2010)
  - SCW: Soft Confidence Weighted (SCW) (Hoi et al., 2012)
- Pros and Cons
  - Faster convergence rate
  - Expensive for high-dimensional data
  - Relatively sensitive to noise

Outline

- Introduction
  - Learning paradigms

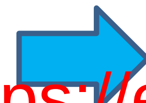
  - Online learning and its applications
- Online learnin

  - Perceptron

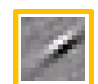
  - Online non-sparse learning
  - Online sparse learning
  - Online unsupervised learning
- Conclusion

Sparse Learning

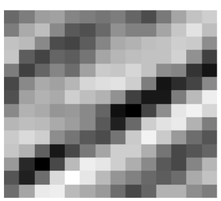
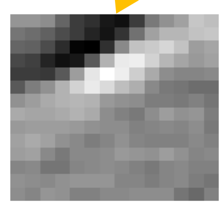
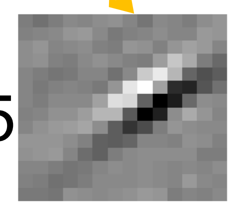
Natural Images



Learned bases ($\phi_{1}, \ldots, \phi_{64}$): "Edges"



Test example

 ≈ 0.8 *  + 0.3 *  + 0.5 

$x \approx 0.8 * \phi_{36} + 0.3 * \phi_{42} + 0.5 * \phi_{63}$

Online Sparse Learning

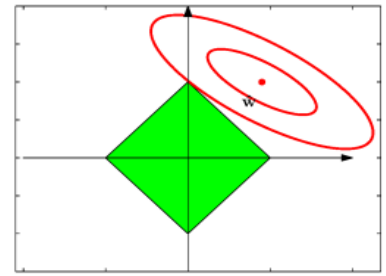
- Motivation
  - Space constraint: RAM overflow
  - Test-time constraint
  - How to induce online learning algorithms?

$$Y = \mathbf{X} \cdot \mathbf{w}$$

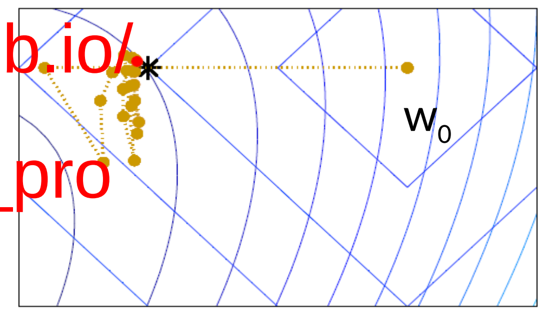
Online Sparse Learning

- Objective function

$$\hat{w} = \arg\min_{w} \sum_{i=1}^{n} L(w, z_i) + g\|w\|_1$$



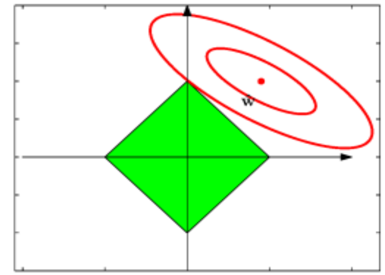
- Problem in online le
  – Standard stochastic



Subgradient

  – It does not yield sparse solution
- Some representative work
  – **Truncated gradient** (Langford et al., 2009)
  – FOBOS: Forward Looking Subgradients (Duchi and Singer, 2009)
  – **Dual averaging** (Xiao, 2009)
  – etc.

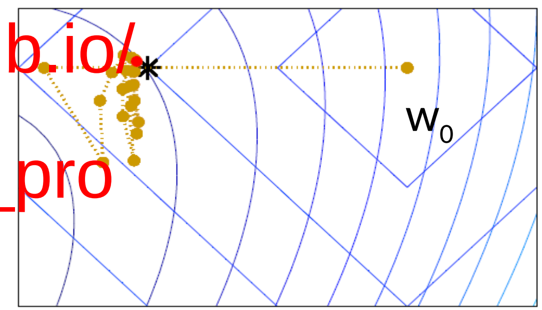
Truncated Gradient (Langford et al., 2009)



- Objective function

$$\hat{w} = \arg\min_{w} \sum_{i=1}^{n} L(w, z_i) + g\|w\|_1$$

- Stochastic gra



Subgradient

- Simple coefficient roundin

$$f(w_i) = T_0(w_i - \eta \nabla_1 L(w_i, z_i), \theta)$$

when the coefficient is small

Truncated gradient: impose sparsity by modifying the stochastic gradient descent

Truncated Gradient (Langford et al., 2009)

Simple Coefficient Rounding vs. Less aggressive truncation

Truncated Gradient (Langford et al., 2009)

$$f(w_i) = T_1(w_i - \eta \nabla_1 L(w_i, z_i), \eta g_i, \theta)$$

- The amount of shrinkage is measured by a *gravity* parameter
- When , the update identical to the sta
- The truncation can be performed every *K* online steps
- Loss functions: $L(w,z) = \phi(w^T x, y)$
  - Logistic $\phi(p,y) = \ln(1+\exp(-py))$
  - SVM (hinge) $\phi(p,y) = \max(0, 1-py)$
  - Least square $\phi(p,y) = \boxed{(p-y)^2}$

Truncated Gradient (Langford et al., 2009)

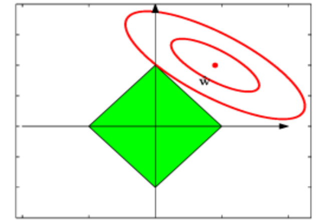
- Theoretical result (T: No. of samples)

- Let , the regret is

$$\sum_{i=1}^{T}(L(w_i,z_i)+g\|w_i\|_1)-\sum_{i=1}^{T}(L(\bar{w},z_i)+g\|\bar{w}\|_1)$$
$$\leq \frac{\sqrt{T}}{2}(B+\|\bar{w}\|^2)\left(1+\frac{A}{2\sqrt{T}}\right)+\frac{A}{2\sqrt{T}}\left(\sum_{i=1}^{T}L(\bar{w},z_i)+g\sum_{i=1}^{T}(\|\bar{w}\|_1-\|w_{i+1}\|_1)\right)+o(\sqrt{T})$$

regret bound is established.

Dual Averaging (Xiao, 2010)



- Objective function

$$\underset{w}{\text{minimize}} \quad \left\{ \phi(w) \triangleq \mathbf{E}_z f(w,z) + \Psi(w) \right\} \quad \Psi(w) = \lambda \| w \|_1 \text{ with } \lambda > 0$$

- Problem: trun n't produce truly sparse weight ng rate

- Fix: dual averaging which o state representations:

  – parameter w_t
  – average gradient vector $\bar{g}_t = \frac{1}{t} \sum_{i=1}^{t} f_i(w_i)$

Dual Averaging (Xiao, 2010)

- Algorithm

- has entry-wise closed-form solution
- Advantage: sparse on the weight
- Disadvantage: keep a non-sparse subgradient

$$w_{t+1}^{(i)} = \begin{cases} 0 & \text{if } \left|\bar{g}_t^{(i)}\right| \leq \lambda, \\ -\frac{\sqrt{t}}{\gamma}\left(\bar{g}_t^{(i)} - \lambda\, \mathrm{sgn}\left(\bar{g}_t^{(i)}\right)\right) & \text{otherwise,} \end{cases}$$

Convergence and Regret

- Average regret

- Theoretical bound: similar nt descent

$$\bar{R}_T \sim \mathcal{O}(1/\sqrt{T})$$

$$\bar{R}_T \sim \mathcal{O}(\log(T)/T), \quad \text{if } h(\cdot) \text{ is strongly convex}$$

average regret bound is established.

Variants of Online Sparse Learning Models

- Online feature selection (OFS)
  - A variant of sparse online learning
  - The key difference is that OFS focuses on selecting a fixed subset of featur ess
  - Could be used a atch feature selection when dealing with big data
- Other existing work
  - Online learning for Group Lasso (Yang et al., 2010) and online learning for multi-task feature selection (Yang et al. 2013) to select features in group manner or features among similar tasks

Online Sparse Learning

- Objective
  - Induce sparsity in the weights of online learning algorithms

- Pros and Con

  - Simple and easy to imple

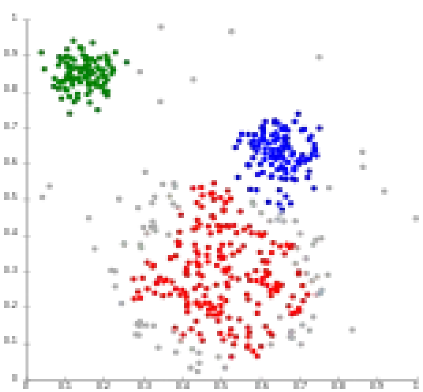
  - Efficient and scalable for high-dimensional data
  - Relatively slow convergence rate
  - No perfect way to attain sparsity solution yet

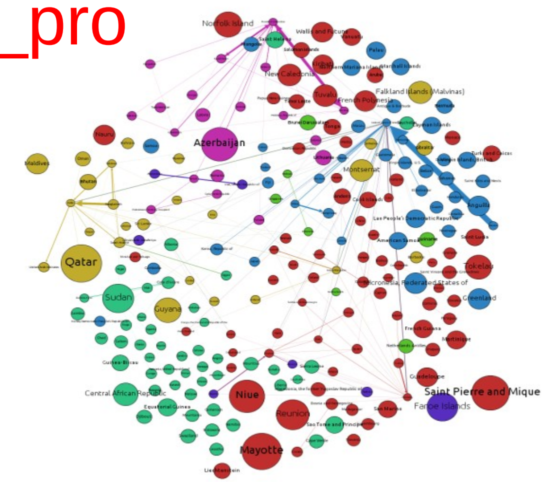
Outline

- Introduction
  - Learning paradigms
  - Online learning and its applications
- Online learning
  - Perceptron
  - Online non-sparse learning
  - Online sparse learning
  - Online unsupervised learning
- Conclusion

Online Unsupervised Learning

- Assumption: data generated from some underlying parametric probabilistic density function
- Goal: estimate the parameters of the density to give a suitable compac



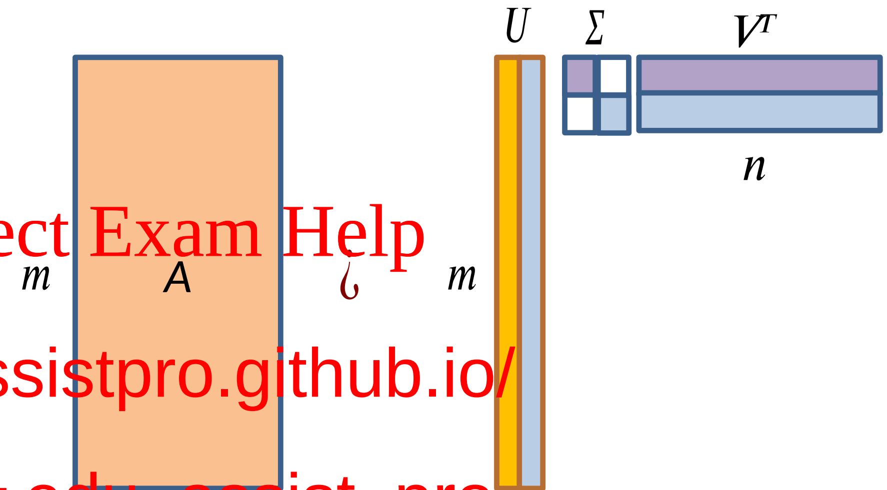


Online Unsupervised Learning

- Some representative work
  - **Online singular value decomposition** (SVD) (Brand, 2003)
  - Online principal component analysis (PCA) (Warmuth and Kuzmin, 2006)
  - Online dictionar ing (Mairal et al. 2009)
  - Online learning for latent Dirich n (LDA) (Hoffman et al., 2010)
  - Online variational inference for the hierarchical Dirichlet process (HDP) (Wang et al. 2011)
  - Online Learning for Collaborative Filtering (Ling et al. 2012)
  - …

SVD: Definition

- : input data matrix
  - matrix (e.g. documents, terms)
- : left singular vectors
  - matrix (documents,
- : singular values
  - diagonal matrix (strength of each "topic")
  - rank of matrix
  - Nonnegative and sorted
- : right singular vectors
  - matrix (terms, topics)

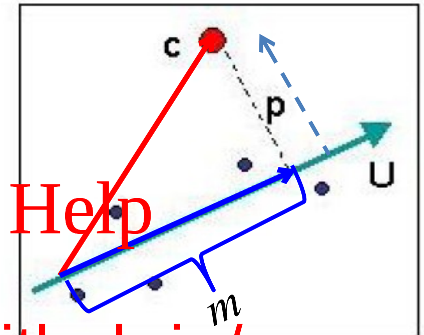


- : scalar
- : vector
- : vector

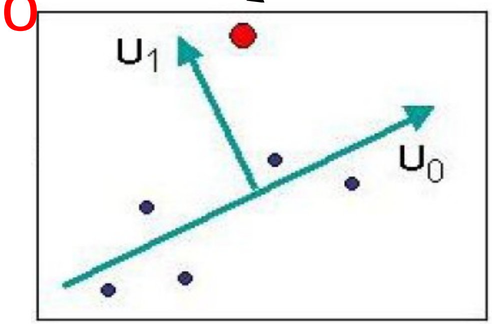
Online SVD (Brand, 2003)

- Challenges: storage and computation
- Idea: an incremental algorithm computes the principal eige without storing the entire ma

Online SVD (Brand, 2003)



$\|p\| < \mathrm{thr}?$



Online SVD (Brand, 2003)

- Complexity

  $O(r^2)$

- Store
  -

- The online SVD has more error, but it is comparable to PCA (SVD)

Online SVD

- Unsupervised learning: minimizing the reconstruction errors
- Each update w by at most one, until a user-sp hed
- Pros and Cons
  - Simple to implement
  - Fast computation
  - Comparable performance
  - Lack of theoretical guarantee

Outline

- Introduction
  - Learning paradigms
  - Online learning and its applications
- Online learnin
  - Perceptron
  - Online non-sparse learning
  - Online sparse learning
  - Online unsupervised learning
- Conclusion

One-slide Takeaway

- Basic concepts
  - What is online learning?
  - What is regret analysis?
- Online learning
  - Perceptron
  - Online gradient descent
  - Passive aggressive
  - Truncated gradient
  - Dual averaging
  - Online SVD

Resources

- Book and Video:
  - Prediction Learning and Games. N. Cesa-Bianchi and G. Lugosi. Cambridge university press, 2006.

  - [Shal11] Online L Optimization. Shai Shalev-Shwartz. F

  Machine Learning, Vol. 4, No. 2, 2011, 107-194. DOI: 10.156 18

  - http://videolectures.net/site/sear e+learning
- Software:
  - Pegasos: http://www.cs.huji.ac.il/~shais/code/index.html
  - VW: hunch.net/~vw/
  - SGD by Leon Bottou: http://leon.bottou.org/projects/sgd

References

- Cesa-Bianchi, Nicol`o, Conconi, Alex, and Gentile, Claudio. A second-order perceptron algorithm. SIAM J. Comput., 34(3):640–668, 2005.
- M. Brand. Fast online svd revisions for lightweight recommender systems. In SDM, 2003.
- N. Cesa-Bianchi, A. Conconi, and C. Gentile. A second-order perceptron algorithm. SIAM J. Comput., 34(3):640–668, 2005.

- K. Crammer, O. Dekel, J. Keshet, ne passive-aggressive algorithms. Journal of Machine Learning Res

- K. Crammer, A. Kulesza, and M. Dredze. Adaptive regula vectors. In NIPS, pages 414–422, 2009.

- K. Crammer and D. D. Lee. Learning via gaussian herding. 1–459, 2010.
- K. Crammer and Y. Singer. Ultraconservative online algorithms for multiclass problems. Journal of Machine Learning Research, 3:951–991, 2003.
- M. Dredze, K. Crammer, and F. Pereira. Confidence-weighted linear classification. In ICML, pages 264–271, 2008.
- C. Gentile. A new approximate maximal margin classification algorithm. Journal of Machine Learning Research, 2:213–242, 2001.
- M. D. Hoffman, D. M. Blei, and F. R. Bach. Online learning for latent dirichlet allocation. In NIPS, pages 856–864, 2010.
- S. C. H. Hoi, J. Wang, and P. Zhao. Exact soft confidence-weighted learning. In ICML, 2012.

References

- J. Langford, L. Li, and T. Zhang. Sparse online learning via truncated gradient. Journal of Machine Learning Research, 10:777–801, 2009.
- Y. Li and P. M. Long. The relaxed online maximum margin algorithm. Machine Learning, 46(1-3):361–387, 2002.

- Guang Ling, Haiqin Yang, Irwin King and M.R. Lyu. Online Learning for Collaborative Filtering, IJCNN, Brisbane, Australia,

- P. L. Lions and B. Mercier. wo nonlinear operators. SIAM J. Numer. Anal., 16(6):964–979, 1979.

- J. Mairal, F. Bach, J. Ponce, and G. Sapiro. Online ng for sparse coding. In ICML, page 87, 2009.
- Y. Nesterov. Gradient methods for minimizing composite objective function. CORE Discussion Paper 2007/76, Catholic University of Louvain, Center for Operations Research and Econometrics, 2007.
- F. Orabona and K. Crammer. New adaptive algorithms for online classification. In NIPS, pages 1840–1848, 2010.
- F. Rosenblatt. The Perceptron: A probabilistic model for information storage and organization in the brain. Psychological Review, 65:386–408, 1958.

References

- C. Wang, J. W. Paisley, and D. M. Blei. Online variational inference for the hierarchical dirichlet process. Journal of Machine Learning Research - Proceedings Track, 15:752–760, 2011.
- M. K. Warmuth and D. Kuzmin. Randomized pca algorithms with regret bounds that are logarithmic in the dimension. In NIPS, pages 1481–1488, 2006.
- S. J. Wright, R. D. Nowak, and M. A. T. Figueiredo. Sparse reconstruction by separable approximation. IEEE Trans 479–2493, 2009.
- L. Xiao. Dual averaging met ing and online optimization. Journal of Machine Learning Research, 11:2543–
- H. Yang, M. R. Lyu, and I. King. Efficient online lea ask feature selection. ACM Transactions on Knowledge Discovery from Data, 2013.
- H. Yang, Z. Xu, I. King, and M. R. Lyu. Online learning for group lasso. In ICML, pages 1191–1198, Haifa, Israel, 2010.
- L. Yang, R. Jin, and J. Ye. Online learning by ellipsoid method. In ICML, page 145, 2009.
- P. Zhao, S. C. H. Hoi, and R. Jin. Duol: A double updating approach for online learning. In NIPS, pages 2259–2267, 2009.
- M. Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In ICML, pages 928–936, 2003.

In-class Practice

- We have two data and , how to get a classifier by Perceptron learning rule?

- Assume
  - is in class (t

  - is in class

- Data points are linearly separable and can be applied repeatedly (for validation).