Machine Learning in Communications
Lecture 2: Role of Machine Learning in Communications

Harpreet S. Dhillon

Wireless@VT, Department of Electrical & Computer Engineering
Virginia Tech, Blacksburg, VA

https://www.dhillon.ece.vt.edu
hdhillon@vt.edu

JTG/IEEE Information Theory Society Summer School
IIT Kanpur

Lecture Objectives

After covering the basics of machine learning in the previous lecture, we will now start our discussion on the role of machine learning in communications.

- Building directly on the previous lecture, we will start with a signal detection example to reinforce the importance of the choice of appropriate loss function and performance measure.
- We will then discuss some potential classes of communications problems that will benefit from machine learning.
- We will conclude with a case study on *Determinantal Learning for Wireless Networks*.

Binary Classification on an Unbalanced Dataset

- Lets assume that each point in our training set has a binary label.
- Assume further that one of the labels occurs very infrequently.
  - Think of a signal detection problem assuming that the message is transmitted very infrequently.
- In many such problems, it is more detrimental if we miss a signal than if we detect a signal that was not there (*false negatives* are more critical than *false positives*).
- Consider the classical example of a medical dataset.
  - Assume that the binary label signifies whether a given patient has a disease or not.
  - It is really critical to detect correctly when a patient has that disease. Otherwise, the treatment may get delayed.
  - On the contrary, if we misclassify a healthy person as having that disease, it is "relatively" easy to handle it (e.g., run more tests).

Binary Classification - Choice of Loss Function

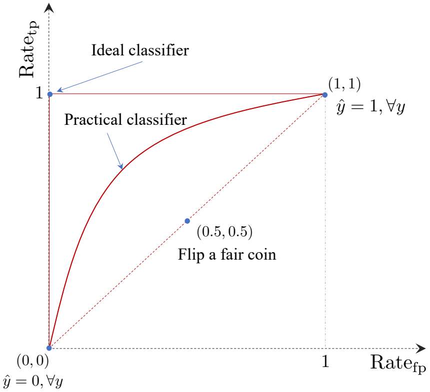
y \ $\hat{y}$	0	1
0	tn 0	fp ?
1	fn ?	tp 0

For the reasons that we already discussed, we may want to put a larger loss for fn. Therefore, simply 0-1 loss function will not work in this case.

Binary Classification - Measuring Accuracy

- Consider a dataset in which only 0.1% of patients have a disease and the rest are healthy. Note that you can easily map this to the signal detection problem as well.
- You propose an algorithm that gives a 99.5% accuracy. Accuracy here is defined as the percentage of points that were correctly classified.
- Is this a good algorithm?
- What about a trivial algorithm that predicts that no one has a disease? In other words, $\hat{y}_i = 0, \forall i$. What is the accuracy of this algorithm?
- Why is this performing better than your algorithm?
- Takeaway: We need to be more careful with how we *measure* accuracy.

Binary Classification - ROC



- Remember the dependence of Rate_{tp} and Rate_{fp} in a signal detection problem on the signal detection threshold.
  - The *practical classifier* curve is obtained by changing this threshold.
- This is called Receiver Operating Characteristics (ROC) curve and is one of the standard tools used in machine learning to characterize the performance of classifiers.

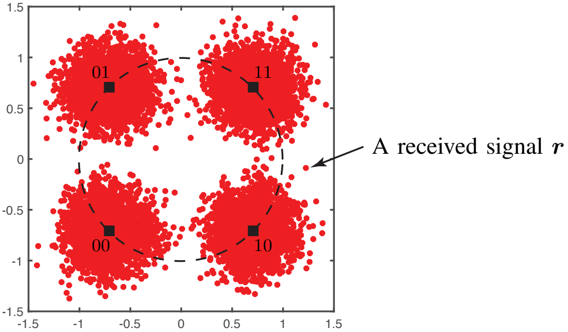
First a Note on Artificial Neural Networks

Communications Problems that can benefit from ML

- Remember from the first lecture that for a ML problem to be meaningful, you need (i) *some* pattern to learn, (ii) data to learn from, and (iii) it should not be possible to describe that pattern mathematically.
- Given this, here are the type of problems that will benefit from ML:
  - Some algorithms may be prohibitively complex for real-time implementation. Can we come up with ML-based solutions? *Case study today*.
  - Mathematical models are inadequate or incomplete to describe the data. *Case study on Day 3*.
  - Data-driven applications, such as edge learning. *Case study on Day 4.*

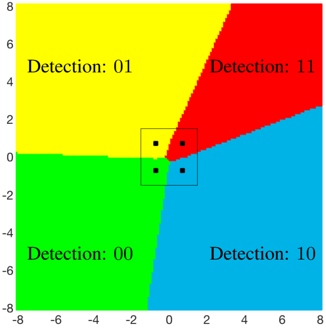
In which problems would ML not make sense? Let's see next.

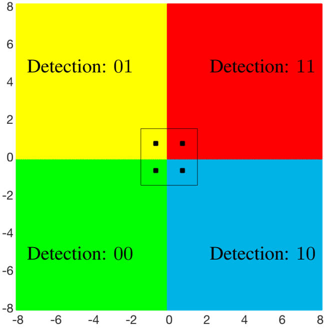
QPSK Example- I



E. Björnson, P. Giselsson, "Two Applications of Deep Learning in the Physical Layer of Communication Systems", *IEEE Signal Processing Magazine*, Sept. 2020.

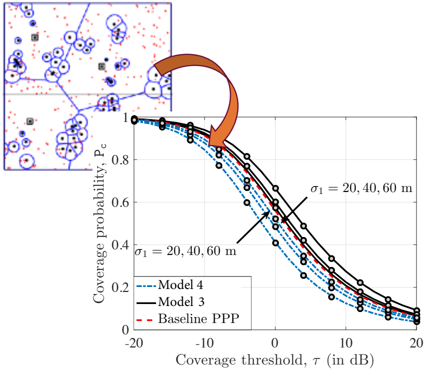
QPSK Example - II

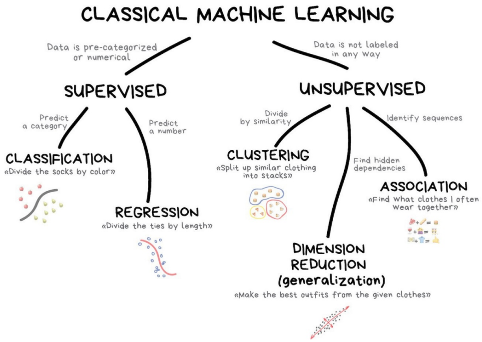




E. Björnson, P. Giselsson, "Two Applications of Deep Learning in the Physical Layer of Communication Systems", *IEEE Signal Processing Magazine*, Sept. 2020.

Case Study: Determinantal Learning for Wireless Networks



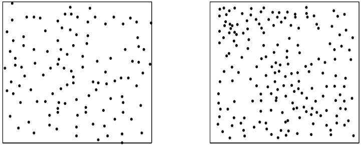


Stochastic geometry is model driven approach, ML is data-driven approach.

C. Saha and H. S. Dhillon, "Machine Learning meets Stochastic Geometry: Determinantal Subset Selection for Wireless Networks", in *Proc. IEEE Globecom*, Waikoloa, HI, Dec. 2019.

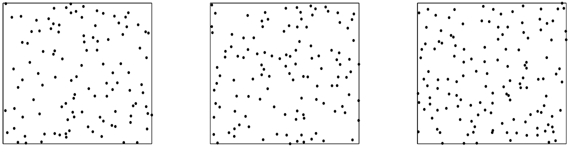
..but there is a connection

- Determinantal point process (DPP): used as a repulsive point process to model the locations of macro base stations in a cellular network*.



(a) Houston data set (b) LA data set

Fig. 1: Real macro BS deployments.

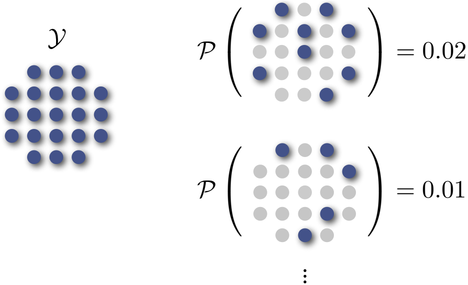


(a) Gauss DPP (b) Cauchy DPP (c) Generalized Gamma DPP

- DPP is also used in ML as a probability model for subset selection.

*Y. Li, F. Baccelli, H. S. Dhillon and J. G. Andrews, "Statistical Modeling and Probabilistic Analysis of Cellular Networks with Determinantal Point Processes", IEEE Trans. on Commun., vol. 63, no. 9, pp. 3405-3422, Sep. 2015

Discrete point processes



- Consider a ground set of N items, $\mathcal{Y} = \{1, 2, \ldots, N\}$
- Power set: $2^{\mathcal{Y}}$
- Any probabilistic subset selection model is a probability measure $\mathcal{P}$ on $2^{\mathcal{Y}}$.

Example: Independent point process

- Each element i is included with probability p_i

$$\mathcal{P}(Y) = \prod_{i \in Y} p_i \prod_{i \notin Y} (1 - p_i)$$

DPP

DPPs are probabilistic models that quantify the likelihood of selecting a subset of items as the determinant of a kernel matrix (K).

Definition 1: K-matrix formulation

- Consider a ground set of N items, $\mathcal{Y} = \{1, 2, \ldots, N\}$
- DPP is a probability measure on the power set $2^{\mathcal{Y}}$
- A random subset $\mathbf{Y}$ follows a DPP if $\mathcal{P}(A \subset \mathbf{Y}) = \det(K_A)$, where $K_A \equiv [K_{i,j}]_{i,j \in A}$
- K is positive semidefinite with $K \preceq I$

L-ensemble formulation of DPP

- A DPP is alternatively defined in terms of a matrix L ($L \preceq I$) indexed by $Y \subseteq \mathcal{Y}$:

$$\mathcal{P}_L(Y) \equiv \mathcal{P}_L(\mathbf{Y} = Y) = \frac{\det(L_Y)}{\det(L + I)},$$

  where $L_Y = [L_{i,j}]_{i,j \in Y}$.
- K and L are related as

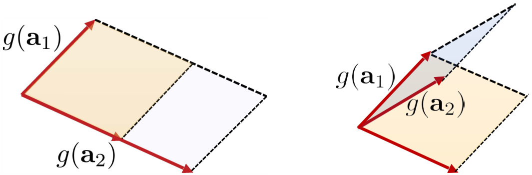
$$K = (L + I)^{-1} L.$$

DPP: Quality-Diversity Tradeoff

- Say, $\mathbf{a}_i \in \mathbb{R}^N$ is some vector representation of the i^{th} item of $\mathcal{Y}$.
- $L_{i,j} = k(\mathbf{a}_i, \mathbf{a}_j) \equiv \phi(\mathbf{a}_i)^\top \phi(\mathbf{a}_j)$
  $k(\cdot,\cdot)$= kernel function, ϕ = feature map.
- **Quality-diversity decomposition:**

$$L_{i,j} = k(\mathbf{a}_i, \mathbf{a}_j) = \underbrace{g(\mathbf{a}_i)}_{\text{quality of } \mathbf{a}_i \ (\forall i \in \mathcal{Y}} \times \underbrace{S_{i,j}}_{\text{similarity of } \mathbf{a}_i \text{ and } \mathbf{a}_j \ (\forall i,j \in \mathcal{Y}, i \neq j)} \times g(\mathbf{a}_j),$$

- $\mathcal{P}_L(\mathbf{Y} = Y) \propto \det(L_Y) = \det(S_Y) \prod_{i \in Y} g(\mathbf{a}_i)^2,$



(a) $g(\mathbf{a}_1)$ increases. (b) $S_{1,2}$ increases.

$\mathcal{P}(Y) = \{1,2\} \propto \mathrm{Vol}(\{\phi(\mathbf{a}_i)\})$: (a) as $g(\mathbf{a}_i)$ increases, the volume increases, (b) as $S_{i,j}$ increases, the volume decreases.

Conditional DPP

X Some external input (say, an image) → $\mathcal{Y}(X)$ Ground set → Conditional DPP $\mathcal{P}(\mathbf{Y} = Y|X) \propto \det(L_Y(X))$ → Y subset

Example: Image search

"New york" →

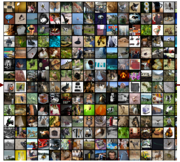
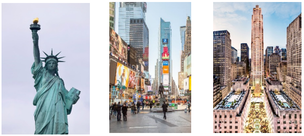


Image library

→ Conditional DPP →



- Similar to the quality-diversity decomposition of DPP,

$$L_{i,j}(X) = \underbrace{g(\mathbf{a}_i|X)}_{\text{quality of item } i \text{ given } X} \times \underbrace{S_{i,j}(X)}_{\text{diversity measure of } i,j \text{ given } X} \times g(\mathbf{a}_j|X).$$

DPP Learning (DPPL) Framework

Setting quality and diversity measures:

- Log-linear model for the quality measure:

$$g(\mathbf{a}_i|X) = \exp\left(\theta^\top \mathbf{f}(\mathbf{a}_i|X)\right),$$

  where $\mathbf{f}$ assigns m feature values to $\mathbf{a}_i$.

- For $S_{i,j}(X)$, we use the Gaussian kernel: $S_{i,j}(X) = e^{-\frac{\|\mathbf{a}_i - \mathbf{a}_j\|^2}{\sigma^2}}$.

Learning setup

- *Training set:* $\mathcal{T} := (X_1, Y_1), \ldots, (X_K, Y_K)$, where X_k is the input and $Y_k \subseteq \mathcal{Y}(X_k)$ is the output.

- $$(\theta^*, \sigma^*) = \arg\max_{(\theta,\sigma)} \mathcal{L}(\mathcal{T}; \theta, \sigma),$$

  where

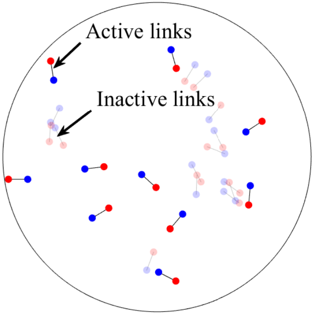
$$\mathcal{L}(\mathcal{T}; \theta, \sigma) = \log \prod_{k=1}^{K} \mathcal{P}_{\theta,\sigma}(Y_k|X_k) = \sum_{k=1}^{K} \log \mathcal{P}_{\theta,\sigma}(Y_k|X_k),$$

  where $\mathcal{P}_{\theta,\sigma} \equiv \mathcal{P}_L$ parameterized by θ and σ.

Case Study: Link Scheduling Problem

System Model

- A network with M Tx-Rx pairs with fixed link distance d.
- Can be represented as a directed bipartite graph $\mathcal{G} := \{\mathcal{N}_t, \mathcal{N}_r, \mathcal{E}\}$,
  - $\mathcal{N}_t$ and $\mathcal{N}_r$ are the independent sets of vertices denoting the set of Tx-s and Rx-s.
  - $\mathcal{E} := \{(t, r)\}$ is the set of directed edges where $t \in \mathcal{N}_t$ and $r \in \mathcal{N}_r$.
- $|\mathcal{N}_t| = |\mathcal{N}_r| = |\mathcal{E}| = M$.



Problem Formulation

- A link is active when the Tx transmits at a power level p_{h} and is inactive when the Tx transmits at a power level p_ℓ (with $0 \leq p_\ell < p_{\mathrm{h}}$).
- Rate on the l^{th} link is given by $\log_2(1 + \gamma_l)$, $\gamma_l = \dfrac{\overbrace{\zeta_{ll}}^{\textbf{Channel gain}} \; p_l}{\underbrace{\sigma^2}_{\textbf{Thermal noise power}} + \sum_{e_j \in \mathcal{E}}^{j \neq l} \zeta_{jl} p_j}$.

Sum-Rate Optimization

- Sum-rate maximization problem:

$$\text{maximize} \sum_{e_l \in \mathcal{E}} \log_2 (1 + \gamma_l),$$

$$\text{subjected to } p_l \in \{p_\ell, p_{\mathrm{h}}\}$$

where $\{p_l\}_{e_l \in \mathcal{E}}$.

- An optimal subset of simultaneously active links : $\mathcal{E}^* \subseteq \mathcal{E}$ [†].

procedure SumRateMax($\mathcal{K}_{\mathcal{N}}^{\mathcal{W}}, \mathcal{E}$)

Initialization: given tolerance $\epsilon > 0$, set $\mathbf{P}_0 = \{p_{l,0}\}$. Set $i = 1$. Compute the initial SINR guess $\hat{\gamma}^{(i)} = \{\gamma_l^{(i)}\}$.

repeat

Solve the GP:

$$\text{minimize } K^{(i)} \prod \gamma_l^{-\frac{\hat{\gamma}_l^{(i)}}{1+\hat{\gamma}_l^{(i)}}} \tag{12a}$$

$$\text{subject to } \beta^{-1}\hat{\gamma}_l^{(i)} \leq \gamma_l \leq \beta\hat{\gamma}_l^{(i)}, e_l \in \mathcal{E}, \tag{12b}$$

$$\sigma^2 \zeta_{ll}^{-1} p_l^{-1} \gamma_l + \sum_{j \neq l} \zeta_{ll}^{-1} \zeta_{jl} p_j p_l^{-1} \gamma_l \leq 1, e_l \in \mathcal{E}, \tag{12c}$$

$$p_l \leq p_{\max}, \ \forall \ e_l \in \mathcal{E}. \tag{12d}$$

with the variables $\{p_l, \gamma_l\}_{e_l \in \mathcal{E}}$. Denote the solution by $\{p_l^*, \gamma_l^*\}_{e_l \in \mathcal{E}}$.

until $\max_{e_l \in \mathcal{E}} |\gamma_l^* - \hat{\gamma}_l^{(i)}| \leq \epsilon$

if $p_l \geq p_{\mathrm{th}}$ **then**

$p_l = p_{\mathrm{h}}$

else

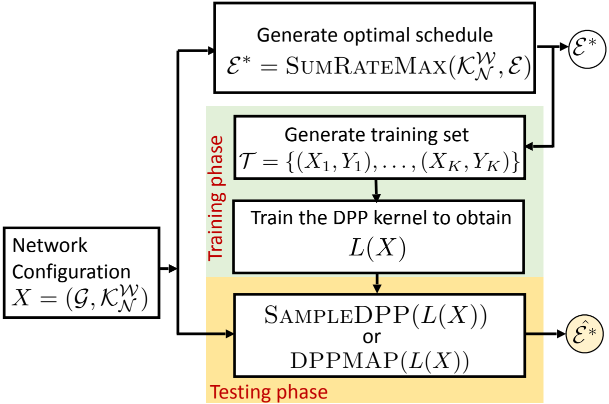
$p_l = p_\ell$

return $\mathcal{E}^*$

- The integer programming problem is NP hard.
- Can be solved iteratively by geometric programming.

[†] P. C. Weeraddana, M. Codreanu, M. Latva-aho, A. Ephremides, C. Fischione et al., "Weighted sum-rate maximization in wireless networks: A review," Foundations and Trends in Networking, vol. 6, no. 1-2, pp. 1-163, 2012.

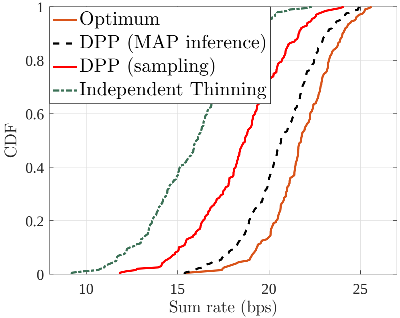
DPPL for Link Scheduling

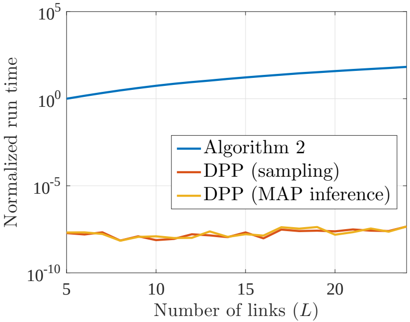


- $\mathcal{K}_{\mathcal{N}_t,\mathcal{N}_r}^{\mathcal{W}}$: complete weighted bipartite graph on $\mathcal{N}_t, \mathcal{N}_r$ with $\mathcal{W}(i,j) = \zeta_{ij}$ for all $i \in \mathcal{N}_t, j \in \mathcal{N}_r$.
- For the training phase, $X_k = (\mathcal{K}_{\mathcal{N}_t,\mathcal{N}_r}^{\mathcal{W}}, \mathcal{E}, \mathcal{E}^*)_k$.

- $g(\mathbf{a}_i|X) := \exp\left(\theta_1 \zeta_{ll} p_{\mathrm{h}} - \theta_2 I_1 - \theta_3 I_2\right)$, where $I_1 = p_{\mathrm{h}} \zeta_{j'i}$ with $j' = \arg\max_{j=1,\ldots,L \neq i}\{\zeta_{ji}\}$ and $I_2 = p_{\mathrm{h}} \zeta_{j''i}$ with $j'' = \arg\max_{j=1,\ldots,L \neq i,j'}\{\zeta_{ji}\}$ are the two strongest interfering powers
- $S_{i,j}(X) = \exp(-(\|\mathbf{x}(t_i) - \mathbf{x}(r_j)\|^2 + \|\mathbf{x}(t_j) - \mathbf{x}(r_i)\|^2)/\sigma^2)$, where $\mathbf{x}(t_i)$ and $\mathbf{x}(r_j)$ denote the locations of Tx $t_i \in \mathcal{N}_t$ and Rx $r_j \in \mathcal{N}_r$.

Results





CDF of sum-rate obtained by different subset selection schemes.

Comparison of run-times of optimal heuristic and DPPL in testing phase.

Trends of Sum-Rate

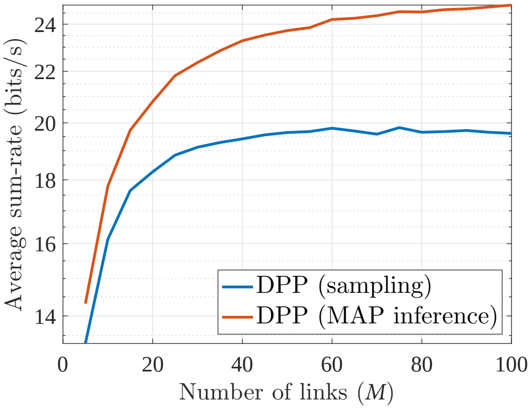


Figure: Average rates obtained for different network sizes using DPPL.

Summary

- Discussed the role of machine learning in communications.
- Today's case study was focused on using machine learning for approximating algorithms.
- We identified a general class of subset selection problems in wireless networks which can be solved by jointly leveraging machine learning and stochastic geometry.
  - Developed the DPPL framework, where the DPP originates from SG and its learning applications have been fine-tuned by the ML community.
  - When applied to a special case of wireless link scheduling, DPP is able to *learn* the underlying quality-diversity tradeoff of the optimal subsets of simultaneously active links.

References

[1] C. Saha and H. S. Dhillon, "Machine Learning meets Stochastic Geometry: Determinantal Subset Selection for Wireless Networks", in *Proc. IEEE Globecom*, Dec. 2019.

[2] B. Blaszczyszyn and P. Keeler, "Determinantal thinning of point pro- cesses with network learning applications," 2018, available online: arXiv/abs/1810.08672.

[3] J. Gillenwater, A. Kulesza, and B. Taskar, "Near-optimal MAP inference for determinantal point processes," in Advances in Neural Information Processing Systems, 2012.

[4] A.Kulesza,B.Taskar et al., "Determinantal point processes for machine learning," Foundations and Trends in Machine Learning, vol. 5, no. 2-3, pp. 123-286, 2012.

[5] C. Saha and H. S. Dhillon, "Interference Characterization in Wireless Networks: A Determinantal Learning Approach", in Proc. IEEE Int. Workshop in Machine Learning for Sig. Processing, Pittsburgh, PA, Oct. 2019.

[6] Y. Li, F. Baccelli, H. S. Dhillon and J. G. Andrews, "Statistical Modeling and Probabilistic Analysis of Cellular Networks with Determinantal Point Processes", IEEE Trans. on Commun., vol. 63, no. 9, pp. 3405-3422, Sept. 2015]

Matlab code available at: https://github.com/stochastic-geometry/DPPL