Learning with Marginalized Corrupted Features

Laurens van der Maaten
Minmin Chen
Stephen Tyree
Kilian Weinberger
Classification of text and image data

Classify....
Classification of text and image data

Classify....

Man Googles Matt Damon's Address Because, Well, He's Crazy And Wants To Murder Him

SALISBURY, MARYLAND—After re-reading actor Matt Damon’s Wikipedia page for the 13th time since 9 a.m. today, local man Dan Easter decided to look up the celebrity’s home address on Google because, well, he’s admittedly crazy and wants to murder him.

Saying he planned to “just click around” a couple websites to see if the Bourne Identity star’s address was listed anywhere on the Internet, Easter told reporters that, you know, he’s ultimately a mentally ill madman who wants to break into Matt Damon’s house in the dead of night and, you guessed it, kill him in front of his wife and children.

“I figured I would just type Matt Damon’s name into Google because, to make a long story short, I’m psychologically disturbed and I want to assassinate him,” said the 29-year-old man, who, by his own admission, is extremely unstable and has absolutely no business being anywhere other than a mental institution. “I see him in movies and magazines all the time, and it made me wonder where he lives. I’m also clinically insane. That’s why

... documents by topic
Classification of text and image data

Classify...

Man Googles Matt Damon's Address Because, Well, He's Crazy And Wants To Murder Him

SALISBURY, MARYLAND—After re-reading actor Matt Damon’s page for the 13th time since 9 a.m. today, local man Dan East look up the celebrity’s home address on Google because, well, admittedly crazy and wants to murder him.

Saying he planned to “just click around” a couple websites to I Bourne Identity star’s address was listed anywhere on the inter told reporters that, you know, he’s ultimately a mentally ill man wants to break into Matt Damon’s house in the dead of night a guessed it, kill him in front of his wife and children.

“I figured I would just type Matt Damon’s name into Google be make a long story short, I’m psychologically disturbed and I w assassinate him,” said the 29-year-old man, who, by his own i extremely unstable and has absolutely no business being any th than a mental institution. “I see him in movies and magazines and it made me wonder where he lives. I’m also clinically ins

Kindle vs. Nook vs. iPad: Which e-book reader should you buy?

With ultra-affordable e-book readers, mid-price color tablets like the Nexus 7 and Kindle Fire, and even the more expensive iPads all vying for your e-book dollar, what’s the best choice for you? It depends.

Editors note, September 7, 2012: As of September 6, Amazon has announced its new Kindle e-readers and tablets for 2012 that dramatically alter the buying decisions listed below. The first wave of the new Amazon products are due to ship by September 14.

We’ll update this story in detail after we review those models. By that time, we’ll also find out what Apple is announcing at its September 12 event. In the meantime, the
Classification of text and image data

Classify....

Man Googles Matt Damon's Address Because, Well, He's Crazy And Wants To Murder Him

SALISBURY, MARYLAND—After re-reading actor Matt Damon’s page for the 13th time since 9 a.m. today, local man Dan East look up the celebrity’s home address on Google because, well admittedly crazy and wants to murder him.

Saying he planned to “just click around” a couple websites to i Bourne Identity star’s address was listed anywhere on the Inter told reporters that, you know, he’s ultimately a mentally ill man who, by his own... wants to break into Matt Damon’s house in the dead of night and guessed it, kill him in front of his wife and children.

“I figured I’d just type Matt Damon’s name into Google be make a long story short, I’m psychologically disturbed and I w assinate him,” said the 25-year-old man, who, by his own... extremely unstable and has absolutely no business being any than a mental institution. I see him in movies and magazines and it made me wonder where he lives. I’m also clinically insan...
Empirical risk minimization

• Learn model based on annotated data $\mathcal{D} = \{(x_n, y_n)\}_{n=1}^{N}$ by minimizing:

$$\min_{\Theta} \mathcal{L}(\mathcal{D}; \Theta) = \sum_{n=1}^{N} L(x_n, y_n; \Theta)$$
Empirical risk minimization

• Learn model based on annotated data $\mathcal{D} = \{(x_n, y_n)\}_{n=1}^{N}$ by minimizing:

$$\min_{\Theta} \mathcal{L}(\mathcal{D}; \Theta) = \sum_{n=1}^{N} L(x_n, y_n; \Theta)$$

• Herein, typical examples of the loss function include:
Regularization by priors

- Regularizers incorporate a term in the loss that penalizes complex models:

\[
\tilde{\mathcal{L}}(x, y; w) = \mathcal{L}(x, y; w) + \lambda \mathcal{R}(w)
\]

\[
e.g., \quad \mathcal{R}(w) = \|w\|^2 \quad \text{or} \quad \mathcal{R}(w) = |w|
\]
Regularization by priors

• Getting the right regularizer is tricky!
Regularization by priors

- Getting the right regularizer is *tricky*!
  - Most norm-based regularizers are rather *arbitrary*:
    - L1 and L2-regularization are popular mainly for computational reasons
Regularization by priors

• Getting the right regularizer is tricky!

  • Most norm-based regularizers are rather arbitrary:
    • L1 and L2-regularization are popular mainly for computational reasons

  • Most practitioners have bad intuitions about model parameters...
    • ... but they do understand their data!
Regularization by priors

- Getting the right regularizer is tricky!

- Most norm-based regularizers are rather arbitrary:
  - L1 and L2-regularization are popular mainly for computational reasons

- Most practitioners have bad intuitions about model parameters...
  - ... but they do understand their data!

- Instead of restricting the model parameters, can’t we incorporate knowledge about the data instead?
Are these reviews positive or negative?

- This is a boring movie with a lot of decadence and bad influence on people. I can't believe this movie won awards! I would not recommend this though it's so famous.

- The movie is great, and in perfect condition. Came in time. I'd recommend the movie itself, and I would purchase movies from here again.

- This movie is awesome, if you have not seen Tarrantino movies on Blu Ray you are missing out. Blu Ray brings these movies to life, especially if you have a good surround sound system.

- I tried to watch. I bought it because of the Micah quote. If you like to watch people get high and talk filthy this is for you.
Regularization by corruption

• Remove each word with probability $q$:

- This is a boring movie with a lot of decadence and bad influence on people. I can't believe this movie won awards! I would not recommend this though it's so famous.
- The movie is great, and in perfect condition. Came in time. I'd recommend the movie itself, and I would purchase movies from here again.
- This movie is awesome, if you have not seen Tarrantino movies on Blu Ray you are missing out. Blu Ray brings these movies to life, especially if you have a good surround sound system.
- I tried to watch. I bought it because of the Micah quote. If you like to watch people get high and talk filthy this is for you.
- This is a fitting movie with a lot of decadence and bad influence on people. I believe this movie won awards! I would not recommend this though it's so famous.
- The movie is great, and in perfect condition. Came in time. I'd recommend the movie itself, and I would purchase movies from here again.
- This movie is awesome, if you have not seen Tarrantino movies on Blu Ray you are missing out. Blu Ray brings these movies to life, especially if you have a good surround sound system.
- I tried to watch. I bought it because of the Micah quote. If you like to watch people get high and talk filthy this is for you.
Regularization by corruption

- Define *label-invariant corruptions* that can be applied to the data

- Training on such corrupted data leads to *robustness* to the corruption

- *Robustness* is intimately related to *regularization* of the model

- We show that this can be done *efficiently* by *marginalizing* over corruptions
Regularization by corruption

• Instead of regularizer, define a *label-invariant corrupting distribution*:

\[
p(\tilde{x}|x) = \prod_{d=1}^{D} p(\tilde{x}_d|x_d; \eta_d), \text{ with } \mathbb{E}[\tilde{x}]p(\tilde{x}|x) = x
\]

• We will assume the corruption are independent across features (this assumption may be relaxed for Gaussian corruptions)
Regularization by corruption

- Instead of regularizer, define a label-invariant corrupting distribution:

\[
p(\tilde{x} | x) = \prod_{d=1}^{D} p(\tilde{x}_d | x_d; \eta_d), \quad \text{with } \mathbb{E}[\tilde{x}]_p(\tilde{x} | x) = x
\]

- We will assume the corruption are independent across features (this assumption may be relaxed for Gaussian corruptions)

<table>
<thead>
<tr>
<th>Distribution</th>
<th>PDF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Blankout noise</td>
<td>( p(\tilde{x}_{nd} = 0) = q_d )</td>
</tr>
<tr>
<td></td>
<td>( p(\tilde{x}<em>{nd} = \frac{1}{1-q_d} x</em>{nd}) = 1-q_d )</td>
</tr>
<tr>
<td>Gaussian noise</td>
<td>( p(\tilde{x}_{nd}</td>
</tr>
<tr>
<td>Laplace noise</td>
<td>( p(\tilde{x}_{nd}</td>
</tr>
<tr>
<td>Poisson noise</td>
<td>( p(\tilde{x}_{nd}</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table 1</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>keep</td>
<td>0</td>
</tr>
<tr>
<td>amazing</td>
<td>7</td>
</tr>
<tr>
<td>ideas</td>
<td>2</td>
</tr>
<tr>
<td>value</td>
<td>0</td>
</tr>
<tr>
<td>poor</td>
<td>0</td>
</tr>
<tr>
<td>average</td>
<td>1</td>
</tr>
</tbody>
</table>
Simple approach

• For each example, generate $M$ corrupted examples and use these as data
• This amounts to minimizing the loss on an augmented, corrupted training set:

$$
\mathcal{L}(\tilde{D}; \Theta) = \sum_{n=1}^{N} \frac{1}{M} \sum_{m=1}^{M} L(\tilde{x}_{nm}, y_n; \Theta) \text{ with } \tilde{x}_{nm} \sim p(\tilde{x}_n | x_n)
$$

This is a boring movie with a lot of decadence and bad influence on people. I can’t believe this movie won awards! I would not recommend this though it’s so famous.
Simple approach

• For each example, generate $M$ corrupted examples and use these as data*

• This amounts to minimizing the loss on an augmented, corrupted training set:

$$\mathcal{L}(\tilde{D}; \Theta) = \sum_{n=1}^{N} \frac{1}{M} \sum_{m=1}^{M} L(\tilde{x}_{nm}, y_n; \Theta) \text{ with } \tilde{x}_{nm} \sim p(\tilde{x}_n|x_n)$$

• This quickly gets computationally prohibitive, unless...
Marginalized Corrupted Features

- For each example, generate $M$ corrupted examples and use these as data.

- This amounts to minimizing the loss on an augmented, corrupted training set:

  $$
  \mathcal{L}(\tilde{D}; \Theta) = \sum_{n=1}^{N} \frac{1}{M} \sum_{m=1}^{M} L(\tilde{x}_{nm}, y_n; \Theta) \text{ with } \tilde{x}_{nm} \sim p(\tilde{x}_n|x_n)
  $$

- This quickly gets computationally prohibitive, unless $M \to \infty$

- Law of large numbers leads to the expected loss under the corruption model:

  $$
  \mathcal{L}(\mathcal{D}; \Theta) = \sum_{n=1}^{N} \mathbb{E}[L(\tilde{x}_n, y_n; \Theta)] p(\tilde{x}_n|x_n)
  $$
Quadratic loss

- Working out the MCF expectation (for independent corruption) gives:

\[
L(D; w) = \sum_{n=1}^{N} \mathbb{E} \left[ \left( w^T \tilde{x}_n - y_n \right)^2 \right]_{p(\tilde{x}_n|x_n)}
\]

\[
= w^T \left( \sum_{n=1}^{N} \mathbb{E}[\tilde{x}_n]\mathbb{E}[\tilde{x}_n]^T + V[\tilde{x}_n] \right) w - 2 \left( \sum_{n=1}^{N} y_n \mathbb{E}[\tilde{x}_n] \right)^T w + N
\]

- Practical if we can compute the \textit{mean} and \textit{variance} of corrupting distribution
Quadratic loss

- Working out the MCF expectation (for independent corruption) gives:

\[
L(D; w) = \sum_{n=1}^{N} E \left[ (w^T \tilde{x}_n - y_n)^2 \right]_{p(\tilde{x}_n|x_n)}
\]

\[
= w^T \left( \sum_{n=1}^{N} E[\tilde{x}_n]E[\tilde{x}_n]^T + V[\tilde{x}_n] \right) w \right) - 2 \left( \sum_{n=1}^{N} y_n E[\tilde{x}_n] \right)^T w + N
\]

- Practical if we can compute the mean and variance of corrupting distribution

- The objective function remains convex; optimal solution given by:

\[
w^* = \left( \sum_{n=1}^{N} E[\tilde{x}_n]E[\tilde{x}_n]^T + V[\tilde{x}_n] \right)^{-1} \left( \sum_{n=1}^{N} y_n E[\tilde{x}_n] \right)
\]
Quadratic loss

- Examples of corrupting distributions of interest:

| Distribution       | PDF                                             | $\mathbb{E}[\tilde{x}_{nd}]p(\tilde{x}_{nd}|x_{nd})$ | $\mathbb{V}[\tilde{x}_{nd}]p(\tilde{x}_{nd}|x_{nd})$ |
|--------------------|-------------------------------------------------|---------------------------------------------------|---------------------------------------------------|
| Blankout noise     | $p(\tilde{x}_{nd} = 0) = q_d$                   | $x_{nd}$                                          | $\frac{1}{1-q_d}x_{nd}^2$                         |
|                    | $p(\tilde{x}_{nd} = \frac{1}{1-q_d}x_{nd}) = 1-q_d$ |                                                   |                                                   |
| Gaussian noise     | $p(\tilde{x}_{nd}|x_{nd}) = \mathcal{N}(\tilde{x}_{nd}|x_{nd},\sigma^2)$ | $x_{nd}$                                          | $\sigma^2$                                        |
| Laplace noise      | $p(\tilde{x}_{nd}|x_{nd}) = Lap(\tilde{x}_{nd}|x_{nd},\lambda)$ | $x_{nd}$                                          | $2\lambda^2$                                      |
| Poisson noise      | $p(\tilde{x}_{nd}|x_{nd}) = Poisson(\tilde{x}_{nd}|x_{nd})$ | $x_{nd}$                                          | $x_{nd}$                                          |

- Using Gaussian corruptions leads to an interesting special case:

$$
\mathcal{L}(\mathcal{D}; \mathbf{w}) = \mathbf{w}^T \left( \sum_{n=1}^{N} \mathbf{x}_n \mathbf{x}_n^T \right) \mathbf{w} - 2 \left( \sum_{n=1}^{N} y_n \mathbf{x}_n \right)^T \mathbf{w} + \sigma^2 N \mathbf{w}^T \mathbf{w} + N
$$

- Minimizing MCF-Gaussian quadratic loss leads to ridge regression!
Exponential loss

• Working out the MCF expectation (for independent corruption) gives:

\[ \mathcal{L}(\mathcal{D}; \mathbf{w}) = \sum_{n=1}^{N} \mathbb{E} \left[ \exp \left( -y_n \mathbf{w}^T \tilde{x}_n \right) \right] p(\tilde{x}_n | x_n) \]

\[ = \sum_{n=1}^{N} \prod_{d=1}^{D} \mathbb{E} \left[ \exp \left( -y_n w_d \tilde{x}_{nd} \right) \right] p(\tilde{x}_{nd} | x_{nd}) \]
Exponential loss

- Working out the MCF expectation (for independent corruption) gives:

\[
\mathcal{L}(\mathcal{D}; \mathbf{w}) = \sum_{n=1}^{N} \mathbb{E} \left[ \exp \left( -y_n \mathbf{w}^T \tilde{x}_n \right) \right] p(\tilde{x}_n | x_n) \\
= \sum_{n=1}^{N} \prod_{d=1}^{D} \mathbb{E} \left[ \exp \left( -y_n w_d \tilde{x}_{nd} \right) \right] p(\tilde{x}_{nd} | x_{nd})
\]

- This can be recognized as a product of moment-generating functions:

\[
M_x(t) = \mathbb{E}[\exp(tx)], \, t \in \mathbb{R}
\]
### Moment-generating functions

Here are some examples of the moment generating function and the characteristic function for comparison. It can be seen that the characteristic function is a Wick rotation of the moment generating function $M_X(t)$ when the latter exists.

<table>
<thead>
<tr>
<th>Distribution</th>
<th>Moment-generating function $M_X(t)$</th>
<th>Characteristic function $\phi(t)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bernoulli $P(X = 1) = p$</td>
<td>$1 - p + pe^t$</td>
<td>$1 - p + pe^{it}$</td>
</tr>
<tr>
<td>Geometric $(1 - p)^{k-1} p$</td>
<td>$\frac{pe^t}{1 - (1 - p)e^t}$ for $t &lt; -\ln(1 - p)$</td>
<td>$\frac{pe^{it}}{1 - (1 - p)e^{it}}$</td>
</tr>
<tr>
<td>Binomial $\text{B}(n, p)$</td>
<td>$(1 - p + pe^t)^n$</td>
<td>$(1 - p + pe^{it})^n$</td>
</tr>
<tr>
<td>Poisson $\text{Pois}(\lambda)$</td>
<td>$e^{\lambda(e^t - 1)}$</td>
<td>$e^{\lambda(e^{it} - 1)}$</td>
</tr>
<tr>
<td>Uniform (continuous) $U(a, b)$</td>
<td>$\frac{e^t - e^{ta}}{t(b - a)}$</td>
<td>$\frac{e^{it} - e^{ita}}{it(b - a)}$</td>
</tr>
<tr>
<td>Uniform (discrete) $U(a, b)$</td>
<td>$\frac{e^{at} - e^{(b+1)t}}{(b - a + 1)(1 - e^t)}$</td>
<td>$\frac{e^{ait} - e^{(b+1)it}}{(b - a + 1)(1 - e^{it})}$</td>
</tr>
<tr>
<td>Normal $\mathcal{N}(\mu, \sigma^2)$</td>
<td>$e^{t\mu + \frac{1}{2}t^2\sigma^2}$</td>
<td>$e^{it\mu - \frac{1}{2}t^2\sigma^2}$</td>
</tr>
<tr>
<td>Chi-squared $\chi^2_k$</td>
<td>$(1 - 2it)^{-k/2}$</td>
<td>$(1 - 2it)^{-k/2}$</td>
</tr>
<tr>
<td>Gamma $\Gamma(k, \theta)$</td>
<td>$(1 + it\theta)^{-k}$</td>
<td>$(1 - it\theta)^{-k}$</td>
</tr>
<tr>
<td>Exponential $\text{Exp}(\lambda)$</td>
<td>$\lambda^{-1}e^{-\lambda t}$</td>
<td>$\lambda^{-1}e^{-it\lambda}$</td>
</tr>
<tr>
<td>Multivariate normal $\mathcal{N}(\mu, \Sigma)$</td>
<td>$e^{tT\mu + \frac{1}{2}t^T\Sigma t}$</td>
<td>$e^{itT\mu - \frac{1}{2}t^T\Sigma t}$</td>
</tr>
<tr>
<td>Degenerate $\delta_a$</td>
<td>$e^{ita}$</td>
<td>$e^{ita}$</td>
</tr>
<tr>
<td>Laplace $L(\mu, b)$</td>
<td>$\frac{e^{t\mu}}{1 - b^2t^2}$</td>
<td>$\frac{e^{it\mu}}{1 + b^2t^2}$</td>
</tr>
<tr>
<td>Negative Binomial $\text{NB}(r, p)$</td>
<td>$\frac{(1 - p)e^{it}}{(1 - pe^{it})^r}$</td>
<td>$\frac{(1 - p)e^{it}}{(1 - pe^{it})^r}$</td>
</tr>
</tbody>
</table>
Blankout: Ensemble interpretation

• MCF with blankout has an interesting interpretation as an ensemble
Blankout: Ensemble interpretation

• MCF with blankout has an interesting interpretation as an ensemble

• Example for model with two input features:

\[
\mathcal{L}(\mathcal{D}; \mathbf{w}) = \sum_{n=1}^{N} \left[ q_1 q_2 + (1 - q_1) q_2 \exp(-y_n w_1 x_{n1}) \\
+ (1 - q_2) q_1 \exp(-y_n w_2 x_{n2}) \\
+ (1 - q_1)(1 - q_2) \exp(-y_n [w_1 x_{n1} + w_2 x_{n2}]) \right]
\]
Blankout: Ensemble interpretation

• MCF with blankout has an interesting interpretation as an ensemble

• Example for model with two input features:

\[
\mathcal{L}(D; w) = \sum_{n=1}^{N} \left[ q_1 q_2 + (1 - q_1) q_2 \exp(-y_n w_1 x_{n1}) \ight. \\
+ (1 - q_2) q_1 \exp(-y_n w_2 x_{n2}) \\
+ (1 - q_1)(1 - q_2) \exp(-y_n [w_1 x_{n1} + w_2 x_{n2}]) \right]
\]
Blankout: Ensemble interpretation

• MCF with blankout has an interesting interpretation as an *ensemble*

• Example for model with two input features:

\[
\mathcal{L}(\mathcal{D}; \mathbf{w}) = \sum_{n=1}^{N} \left[ q_1 q_2 + (1 - q_1) q_2 \exp(-y_n w_1 x_{n1}) \\
+ (1 - q_2) q_1 \exp(-y_n w_2 x_{n2}) \\
+ (1 - q_1)(1 - q_2) \exp(-y_n [w_1 x_{n1} + w_2 x_{n2}]) \right]
\]

Loss on first feature subset
Loss on second feature subset
Blankout: Ensemble interpretation

- MCF with blankout has an interesting interpretation as an ensemble

- Example for model with two input features:

\[
\mathcal{L}(\mathcal{D}; w) = \sum_{n=1}^{N} \left[ q_1 q_2 + (1 - q_1)q_2 \exp(-y_n w_1 x_{n1}) + (1 - q_2)q_1 \exp(-y_n w_2 x_{n2}) + (1 - q_1)(1 - q_2) \exp(-y_n [w_1 x_{n1} + w_2 x_{n2}]) \right]
\]

- Loss on first feature subset
- Loss on second feature subset
- Loss on full feature set
Blankout: Ensemble interpretation

• MCF with blankout has an interesting interpretation as an ensemble

• Example for model with two input features:

\[ \mathcal{L}(D; w) = \sum_{n=1}^{N} \left[ q_1 q_2 + (1 - q_1) q_2 \exp(-y_n w_1 x_{n1}) + (1 - q_2) q_1 \exp(-y_n w_2 x_{n2}) + (1 - q_1)(1 - q_2) \exp(-y_n [w_1 x_{n1} + w_2 x_{n2}]) \right] \]

• Note: MCF exponential loss is convex for all corrupting distributions
Logistic loss

• Working out the MCF expectation (for independent corruption) gives:

\[ \mathcal{L}(\mathcal{D}; w) = \sum_{n=1}^{N} \mathbb{E} \left[ \log \left( 1 + \exp \left( -y_n w^T \tilde{x}_n \right) \right) \right] p(\tilde{x}_n | x_n) \]

\[ \leq \sum_{n=1}^{N} \log \left( 1 + \prod_{d=1}^{D} \mathbb{E} \left[ \exp \left( -y_n w_d \tilde{x}_{nd} \right) \right] p(\tilde{x}_{nd} | x_{nd}) \right) \]

• The upper bound is obtained using Jensen's inequality*

* Jensen's inequality: \( \mathbb{E}[\phi(x)] \geq \phi(\mathbb{E}[x]) \) for convex \( \phi(x) \)
Logistic loss

• Working out the MCF expectation (for independent corruption) gives:

\[
\mathcal{L}(\mathcal{D}; \mathbf{w}) = \sum_{n=1}^{N} \mathbb{E} \left[ \log \left( 1 + \exp \left( -y_n \mathbf{w}^T \tilde{x}_n \right) \right) \right]_{p(\tilde{x}_n | x_n)} \\
\leq \sum_{n=1}^{N} \log \left( 1 + \prod_{d=1}^{D} \mathbb{E} \left[ \exp \left( -y_n w_d \tilde{x}_{nd} \right) \right]_{p(\tilde{x}_{nd} | x_{nd})} \right)
\]

• The upper bound is obtained using \textit{Jensen’s inequality}*

• Upper bound is \textit{convex} iff the moment-generating function is \textit{log-linear}

* \textit{Jensen’s inequality}: \( \mathbb{E}[\phi(x)] \geq \phi(\mathbb{E}[x]) \) for convex \( \phi(x) \)
Using MCF in practice

1) Quadratic loss

2) Blankout noise

3) Poisson noise

4) Gaussian noise

5) Exponential loss

6) Logistic loss

\[ \mathcal{L}(\mathcal{D}; w) \]
Experimental setup

- We performed three sets of experiments with MCF:
  - Document classification based on bag-of-word features
  - Image classification based on bag-of-visual-word features
  - “Nightmare at test time” scenario where features are unobserved at test time

- All our predictors use L2-regularization, with lambda set by cross-validation
Experiment 1: Document classification

• We tested on three different document classification data sets

• All data sets have in the order of 20K features and 6K training examples
Experiment 1: Document classification

• We tested on three different document classification data sets

• All data sets have in the order of 20K features and 6K training examples

• We explore two different corrupting distributions:

  • Blankout corruption: 
    \[ p(\tilde{x}_{nd} = 0) = q_d \]
    \[ p(\tilde{x}_{nd} = \frac{1}{1-q_d} x_{nd}) = 1 - q_d \]

  • Poisson corruption: 
    \[ p(\tilde{x}_{nd}|x_{nd}) = Pois(\tilde{x}_{nd}|x_{nd}) \]
Experiment 1: Document classification

The results show: (1) that MCF consistently improves over the blankout corruption level. In the case of MCF with blankout corruption and no additional hyperparameters. In the Amazon data set, we observe that the optimal level of regularization becomes superfluous. Hence, MCF appears to decrease the tendency of predictors to overfit, as a result of the blankout corruption level. In many of the experiments with Poisson corruption (for all corruption levels), we assume that MCF-trained losses (in particular, when blankout corruption is used), we observe that the optimal level of regularization is 0. Hence, MCF appears to decrease the tendency of predictors to overfit, as a result of the blankout corruption level. In the order of 0\%.

MCF significantly outperforms their counterparts without MCF. In the order of 0\%. By contrast, on the Amazon data if the predictors that do not employ MCF, we assume that the predictors that do not employ MCF are properly regularized nor corrupted using MCF. In our experiments on the Dmoz and Reuters data sets, we investigate the performance of MCF as a function of the corruption level. In the order of 0\%.

In all experiments, the exponential and logistic losses is performed by running cross-validate over the blankout corruption parameter (where appropriate). By contrast, on the Amazon data set, we investigate the performance of MCF as a function of the corruption level.

Classification errors of MCF predictors using blankout and Poisson corruption – as a function of the blankout corruption level. The minimization of the (expected) exponential, and logistic loss functions. In all experiments, the validation. The case of MCF with blankout corruption and Poisson corruption (for all corruption levels).

Classification errors are represented on the y-axis, whereas the blankout corruption level is represented on the x-axis.
Experiment 1: Document classification

- Comparing *explicit* and *implicit* blankout corruption (Amazon Books; quadratic loss):

![Graph showing classification error against the number of corrupted copies. The graph compares explicit and implicit corruption (MCF).]
Experiment 2: Image classification

- The CIFAR-10 data set contains 50K images of size 32x32 with 10 classes
- We use a standard* bag-of-visual-words feature representation for the images

<table>
<thead>
<tr>
<th>Experiment 2: Image classification</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>airplane</strong></td>
</tr>
<tr>
<td><strong>automobile</strong></td>
</tr>
<tr>
<td><strong>bird</strong></td>
</tr>
<tr>
<td><strong>cat</strong></td>
</tr>
<tr>
<td><strong>deer</strong></td>
</tr>
<tr>
<td><strong>dog</strong></td>
</tr>
<tr>
<td><strong>frog</strong></td>
</tr>
<tr>
<td><strong>horse</strong></td>
</tr>
<tr>
<td><strong>ship</strong></td>
</tr>
<tr>
<td><strong>truck</strong></td>
</tr>
</tbody>
</table>

* We followed the approach by Coates et al. (2011) to extract features.

<table>
<thead>
<tr>
<th></th>
<th>Quadr.</th>
<th>Expon.</th>
<th>Logist.</th>
</tr>
</thead>
<tbody>
<tr>
<td>No MCF</td>
<td>32.6%</td>
<td>39.7%</td>
<td>38.0%</td>
</tr>
<tr>
<td>Poisson MCF</td>
<td>29.1%</td>
<td>39.5%</td>
<td>30.0%</td>
</tr>
<tr>
<td>Blankout MCF</td>
<td>32.3%</td>
<td>37.9%</td>
<td>29.4%</td>
</tr>
</tbody>
</table>
Experiment 3: “Nightmare at test time”

• In some learning settings, features may be randomly unobserved at test time.

• We experiment with this “nightmare at test time” scenario on MNIST digits:

  • Train regular and MCF-blankout classifiers on the original training set.

```
3 6 8 1 7 9 6 6 9 1
6 7 5 7 8 6 3 4 8 5
2 1 7 9 7 1 2 4 4 5
4 8 1 9 0 1 8 8 9 4
7 6 1 8 6 4 1 5 6 0
7 5 9 2 6 5 8 1 9 7
2 2 2 2 2 3 4 4 8 0
0 2 3 8 0 7 3 8 5 7
0 1 4 6 4 6 0 2 4 3
7 1 2 8 9 6 9 8 6 1
```
Experiment 3: “Nightmare at test time”

• In some learning settings, features may be randomly unobserved at test time

• We experiment with this “nightmare at test time” scenario on MNIST digits:
  
  • Train regular and MCF-blankout classifiers on the original training set
  
  • Randomly delete features from the test images, and measure classification error
Experiment 3: “Nightmare at test time”

- Classification error on test images with randomly deleted features:
Conclusions

- Adding *corrupted* examples to training data can *regularize* predictors
Conclusions

• Adding *corrupted* examples to training data can *regularize* predictors

• For a range of models and corrupting distributions, MCF makes this *efficient*
Conclusions

• Adding *corrupted* examples to training data can *regularize* predictors

• For a range of models and corrupting distributions, MCF makes this *efficient*

• MCF may lead to *improved results* in various learning settings:
  
  • In particular, in settings where you somewhat understand *how* data is generated
  
  • MCF may be very well suited for scenarios in which *domain shift* is present
Thank you! Questions?

Thanks to:

Kilian Weinberger
Minmin Chen
Stephen Tyree