Math 189r
Homework 3
November 21, 2016

There are 5 problems in this set. You need to do 3 problems the first week and 2 the second week. Instead of a sixth problem, **you are encouraged to work on your final project**. Feel free to work with other students, but make sure you write up the homework and code on your own (no copying homework *or* code; no pair programming). Feel free to ask students or instructors for help debugging code or whatever else, though. When implementing algorithms you may not use any library (such as `sklearn`) that already implements the algorithms but you may use any other library for data cleaning and numeric purposes (`numpy` or `pandas`). Use common sense. Problems are in no specific order.

1 (Gaussian Mixture Model) Consider the generative process for a Gaussian Mixture Model:

(1) Draw $z_i \sim \text{Cat}(\boldsymbol{\pi})$ for $i = 1, 2, \ldots, n$.

(2) Draw $\mathbf{x}_i \sim \mathcal{N}(\boldsymbol{\mu}_{z_i}, \boldsymbol{\Sigma}_{z_i})$ for $i = 1, 2, \ldots, n$.

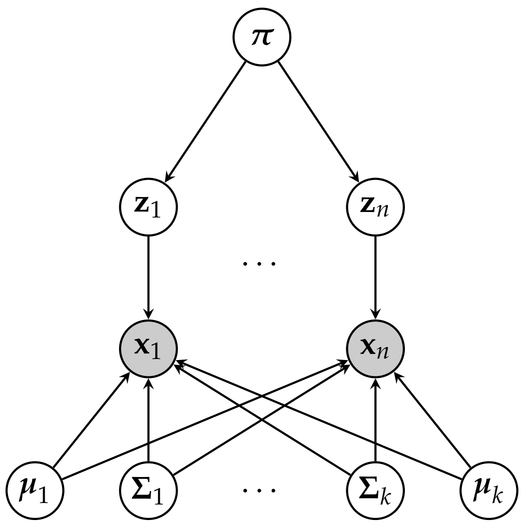
Note that z_i is unobserved but $\mathbf{x}_i$ is observed. Express this model as a directed graphical model, first 'unrolled' and then using Plate notation, before answering the following questions. Support all claims.

(a) Is $\boldsymbol{\pi}$ independent of $\boldsymbol{\mu}_{z_i}$ or $\boldsymbol{\Sigma}_{z_i}$ given your dataset $\mathcal{D} = \{\mathbf{x}_i\}$? Does the posterior distribution over $\{\boldsymbol{\mu}, \boldsymbol{\Sigma}\}$ and $\boldsymbol{\pi}$ factorize? How does this change what inference procedure we need to use for this model?

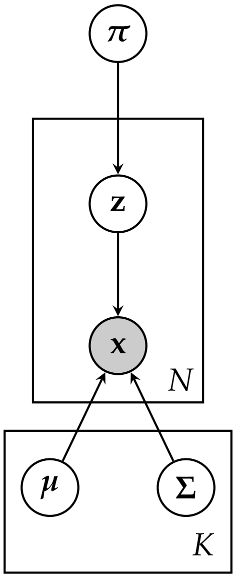
(b) If z_i were observed, would this change? Would the posterior then factorize? *Hint:* what other model have we studied that corresponds to observing z_i?

(c) Find the maximum likelihood estimates for $\boldsymbol{\pi}$, $\boldsymbol{\mu}_k$, and $\boldsymbol{\Sigma}_k$ *if* the latent variables z_i were observed.

Unrolled



Plate



(a) Using the Bayes' Ball algorithm we can see that by d-separation $\boldsymbol{\pi}$ and $\boldsymbol{\mu}$ or $\boldsymbol{\Sigma}$ are conditionally dependent given $\mathcal{D}$. This implies that the posterior won't factorize. Since the posterior doesn't factorize, we aren't able to easily find a maximum likelihood estimate in closed form, and hence we are pushed into using the EM algorithm or some more complicated optimization procedure.

(b) By d-separation, we can see that conditioning on z blocks dependence between $\boldsymbol{\pi}$ and $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$. This implies that the posterior distribution will factorize between $\boldsymbol{\pi}$ and $\{\boldsymbol{\mu}, \boldsymbol{\Sigma}\}$. Note that this model corresponds to Gaussian Discriminant Analysis.

(c) From our results with Gaussian Discriminant Analysis we can see that

$$\boldsymbol{\pi}_k^\star = \frac{\sum_i \mathbf{1}\{z_i = k\}}{N} \tag{1}$$

$$\boldsymbol{\mu}_k^\star = \frac{1}{\sum_i \mathbf{1}\{z_i = k\}} \sum_i \mathbf{x}_i \mathbf{1}\{z_i = k\} \tag{2}$$

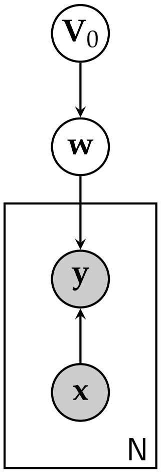
$$\boldsymbol{\Sigma}_k^\star = \sum_i (\mathbf{x}_i - \boldsymbol{\mu}_k)(\mathbf{x}_i - \boldsymbol{\mu}_k)\mathbf{1}\{z_i = k\}. \tag{3}$$

2 (Linear Regression) Consider the Bayesian Linear Regression model with the following generative process:

(1) Draw $\mathbf{w} \sim \mathcal{N}(\mathbf{0}, \mathbf{V}_0)$

(2) Draw $\mathbf{y}_i \sim \mathcal{N}(\mathbf{w}^\top \mathbf{x}_i, \sigma^2)$ for $i = 1, 2, \ldots, n$ where σ^2 is known.

Express this model as a directed graphical model using Plate notation. Is $\mathbf{y}_i$ independent of $\mathbf{w}$? Is $\mathbf{y}_i$ independent of $\mathbf{w}$ *given* $\mathcal{D} = \{\mathbf{x}_i\}$? Support these claims.



By *d*-separation we can see that $\mathbf{y}$ is *not* independent of $\mathbf{w}$ regardless of whether we condition on $\mathbf{x}$ because $\mathbf{y}$ is an observed child of $\mathbf{w}$, effectively observing $\mathbf{w}$. Note that this would change if we are predicting an unobserved $\mathbf{y}$, but that is a different model!

3 (Collaborative Filtering) Consider the 'ratings' matrix $\mathbf{R} \in \mathbb{R}^{m \times n}$ with the low rank approximation $\mathbf{R} = \mathbf{U}\mathbf{V}^\top$ where $\mathbf{U} \in \mathbb{R}^{m \times k}$ and $\mathbf{V} \in \mathbb{R}^{n \times k}$ with k latent factors. Define our optimization problem as

$$\text{minimize: } f(\mathbf{U}, \mathbf{V}) = \|\mathbf{R} - \mathbf{U}\mathbf{V}^\top\|_2^2 + \lambda\|\mathbf{U}\|_2^2 + \gamma\|\mathbf{V}\|_2^2$$

where $\|\cdot\|_2$ in this case is the Frobenius norm $\|\mathbf{R}\|_2^2 = \sum_{ij} \mathbf{R}_{ij}^2$. Derive the gradient of f with respect to $\mathbf{U}_i$ and $\mathbf{V}_j$. Derive a stochastic approximation to this gradient where you consider a single data point at a time.

We can see that

$$\begin{aligned} f(\mathbf{U}, \mathbf{V}) &= \sum_{ij} (\mathbf{R}_{ij} - \mathbf{U}_i \mathbf{V}_j^\top)^2 + \lambda \sum_i \mathbf{U}_i \mathbf{U}_i^\top + \gamma \sum_j \mathbf{V}_j \mathbf{V}_j^\top \qquad (4) \\ &= \sum_{ij} (r_{ij} - \mathbf{u}_i^\top \mathbf{v}_j)^2 + \lambda \sum_i \mathbf{u}_i^\top \mathbf{u}_j + \gamma \sum_j \mathbf{v}_j^\top \mathbf{v}_j. \qquad (5) \end{aligned}$$

This gives

$$\begin{aligned} \nabla_{\mathbf{u}_i} f &= -2 \sum_j (r_{ij} - \mathbf{u}_i^\top \mathbf{v}_j)\mathbf{v}_j + 2\lambda \mathbf{u}_j \qquad (6) \\ \nabla_{\mathbf{v}_j} f &= -2 \sum_i (r_{ij} - \mathbf{u}_i^\top \mathbf{v}_j)\mathbf{u}_i + 2\gamma \mathbf{v}_j \qquad (7) \end{aligned}$$

Since $\mathbf{U}_i = \mathbf{u}_i^\top$ and $\mathbf{V}_j = \mathbf{v}_j^\top$, we have

$$\begin{aligned} \nabla_{\mathbf{U}_i} f &= -2 \sum_j (r_{ij} - \mathbf{U}_i \mathbf{V}_j^\top)\mathbf{V}_j + 2\lambda \mathbf{U}_i \qquad (8) \\ \nabla_{\mathbf{V}_j} f &= -2 \sum_i (r_{ij} - \mathbf{U}_i \mathbf{V}_j^\top)\mathbf{U}_i + 2\gamma \mathbf{V}_j. \qquad (9) \end{aligned}$$

For the stochastic approximation, note that the expected value of

$$f(\mathbf{U}, \mathbf{V}) = (r_{ij} - \mathbf{U}_i \mathbf{V}_j^\top)^2 + \lambda \sum_i \mathbf{U}_i \mathbf{U}_i^\top + \gamma \sum_j \mathbf{V}_j \mathbf{V}_j^\top \qquad (r_{ij} \sim \text{Unif}(\mathbf{R}))$$

is the same as the scaled objective

$$f(\mathbf{U}, \mathbf{V}) = \frac{1}{|\mathbf{R}|} \sum_{ij} (r_{ij} - \mathbf{U}_i \mathbf{V}_j^\top)^2 + \lambda \sum_i \mathbf{U}_i \mathbf{U}_i^\top + \gamma \sum_j \mathbf{V}_j \mathbf{V}_j^\top. \qquad (|\mathbf{R}| \text{ is cardinality})$$

So when $r_{ij} \sim \text{Unif}(\mathbf{R})$ this furnishes the stochastic gradient

$$\begin{aligned} \nabla_{\mathbf{U}_i} f &= -2(r_{ij} - \mathbf{U}_i \mathbf{V}_j^\top)\mathbf{V}_j + 2\lambda \mathbf{U}_i \qquad (10) \\ \nabla_{\mathbf{V}_j} f &= -2(r_{ij} - \mathbf{U}_i \mathbf{V}_j^\top)\mathbf{U}_i + 2\gamma \mathbf{V}_j \qquad (11) \end{aligned}$$

4 (Alternating Least Squares) Consider the same setup and objective

$$\text{minimize: } f(\mathbf{U}, \mathbf{V}) = \|\mathbf{R} - \mathbf{U}\mathbf{V}^\top\|_2^2 + \lambda\|\mathbf{U}\|_2^2 + \gamma\|\mathbf{V}\|_2^2$$

as above in problem (3).

(a) Suppose we fix **U**. Find the optimal **V**.

(b) Suppose we fix **V**. Find the optimal **U**.

(c) Propose an EM-like (block coordinate ascent, to be exact) algorithm for minimizing $f(\mathbf{U}, \mathbf{V})$ using (a) and (b).

(d) Will the algorithm you propose in (c) necessarily converge to the global optimal?

(a) Note that given the form of the Frobenius norm the objective is equivalent (to a constant) to

$$f(\mathbf{U}, \mathbf{V}) = \sum_j \|\mathbf{R}_{:j} - \mathbf{U}\mathbf{V}_j^\top\|_2^2 + \gamma\|\mathbf{V}_j\|_2^2, \tag{12}$$

a factorized objective over each row of **V**. Since this is of the normal Ridge Regression form we see that the optimal

$$\mathbf{V}_j^\star = (\mathbf{U}^\top\mathbf{U} + \gamma\mathbf{I})^{-1}\mathbf{U}^\top\mathbf{R}_{:j}. \tag{13}$$

This gives

$$\mathbf{V}^\star = (\mathbf{U}^\top\mathbf{U} + \gamma\mathbf{I})^{-1}\mathbf{U}^\top\mathbf{R}. \tag{14}$$

(b) We similarly see that to a constant

$$\begin{aligned} f(\mathbf{U}, \mathbf{V}) &= \sum_i \|\mathbf{R}_i - \mathbf{U}_i\mathbf{V}^\top\|_2^2 + \lambda\|\mathbf{U}_i\|_2^2 & (15) \\ &= \sum_i \|\mathbf{R}_i^\top - \mathbf{V}\mathbf{U}_i^\top\|_2^2 + \lambda\|\mathbf{U}_i\|_2^2. & (16) \end{aligned}$$

As this is almost equivalent to the form in part (a) we have

$$\mathbf{U}^\star = (\mathbf{V}^\top\mathbf{V} + \gamma\mathbf{I})^{-1}\mathbf{V}^\top\mathbf{R}^\top. \tag{17}$$

(c) We use a block coordinate descent algorithm[1].

(d) The algorithm won't necessarily converge to the global optimal because even the one dimensional unregularized case minimize : $f(u, v) = (r - uv)^2$ has Hessian

$$\nabla^2 f = \begin{bmatrix} 2v^2 & 4uv - 2r \\ 4uv - 2r & 2u^2 \end{bmatrix}. \tag{18}$$

[1] `http://stanford.edu/~boyd/cvxbook/`

```
**Alternating Least Squares**
input  : instantiated matrices $\mathbf{U}_0$ and $\mathbf{V}_0$, ratings matrix $\mathbf{R}$, regularization parameters
         $\lambda$ and $\gamma$, tolerance $\epsilon$
output: locally optimal $\mathbf{U}^\star$ and $\mathbf{V}^\star$
$t \leftarrow 0$
while $\|\mathbf{U}_t - \mathbf{U}_{t-1}\|_2^2 \geq \epsilon$ and $\|\mathbf{V}_t - \mathbf{V}_{t-1}\|_2^2 \geq \epsilon$ do
    $\mathbf{U}_{t+1} \leftarrow (\mathbf{V}_t^\top \mathbf{V}_t + \gamma \mathbf{I})^{-1} \mathbf{V}_t^\top \mathbf{R}^\top$
    $\mathbf{V}_{t+1} \leftarrow (\mathbf{U}_t^\top \mathbf{U}_t + \gamma \mathbf{I})^{-1} \mathbf{U}_t^\top \mathbf{R}$
    $t \leftarrow t + 1$
end
return  $\mathbf{U}^\star = \mathbf{U}_{t+1}$, $\mathbf{V}^\star = \mathbf{V}_{t+1}$
```

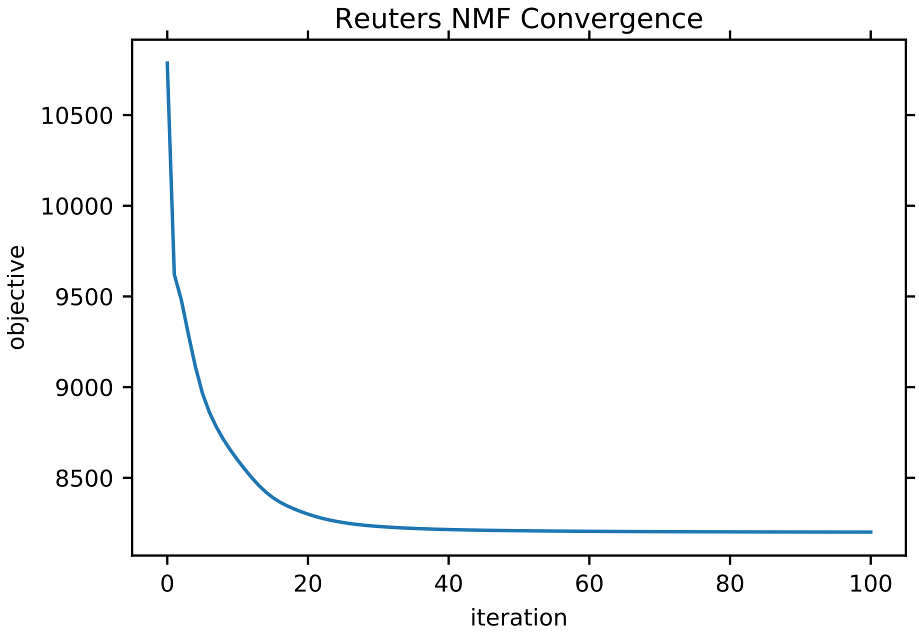
This gives

$$|\nabla^2 f| = -4(r - 3uv)(r - uv), \tag{19}$$

which is less than 0 when $r > 0$ and $u = v = 0$. It follows that the objective is not convex for the most simple case, which generalizes to the larger case. Since the objective is not convex there can exist multiple locally optimal solutions.

5 (Non-Negative Matrix Factorization) Consider the dataset at `http://kdd.ics.uci.edu/databases/reuters21578/reuters21578.html`. Choosing an appropriate objective function and algorithm from Lee and Seung 2001[2] implement Non-Negative Matrix Factorization for topic modelling (choose an appropriate number of topics/latent features) and assert that the convergence properties proved in the paper hold. Display the 20 most relevant words for each of the topics you discover.

We chose to minimize the Frobenius norm between the approximation and the actual matrix, but it's equally okay to minimize the fake KL divergence objective they present in the paper. We also chose to use the multiplicative update because it's easier to implement and guarentees monotonicity of the objective (which we check below with the convergence plot). Also below are the most important words for each of the 20 topics. Note that you might have something different since the objective is non-convex, though they should still look coherent.



```
topic 0: ["billion" "surplus" "deficit" "francs" "marks" "in" "reserves" "deposits"
 "account" "from" "rose" "fell" "assets" "loans" "january" "current" "to"
 "dlrs" "1986" "trade"]
topic 1: ["tonnes" "wheat" "sugar" "corn" "87" "export" "for" "to" "grain" "of" "ec"
 "at" "usda" "1986" "tender" "tonne" "traders" "china" "maize" "exports"]
topic 2: ["cts" "vs" "qtr" "shr" "1st" "net" "inc" "sales" "4th" "2nd" "lt" "28"
 "corp" "31" "jan" "feb" "six" "note" "share" "30"]
topic 3: ["pct" "in" "february" "january" "rose" "year" "rise" "from" "rate" "index"
 "1986" "december" "prices" "fell" "inflation" "compared" "after"
```

[2] `https://papers.nips.cc/paper/1861-algorithms-for-non-negative-matrix-factorization.pdf`

```
 "statistics" "consumer" "growth"]
topic 4: ["stg" "bank" "money" "market" "england" "bills" "band" "assistance" "the"
 "of" "shortage" "at" "today" "revised" "help" "around" "forecast" "rate"
 "pct" "central"]
topic 5: ["vs" "net" "revs" "shr" "mths" "nine" "3rd" "cts" "dlrs" "qtr" "lt" "note"
 "mln" "12" "includes" "corp" "nil" "31" "name" "26"]
topic 6: ["mln" "vs" "1986" "tax" "stg" "11" "28" "16" "sales" "extraordinary" "13"
 "17" "pretax" "29" "27" "12" "37" "note" "turnover" "15"]
topic 7: ["000" "net" "sales" "includes" "note" "vs" "500" "cts" "700" "100" "gain"
 "credits" "of" "600" "tax" "20" "200" "slaughter" "periods" "and"]
topic 8: ["the" "to" "he" "that" "said" "would" "in" "and" "be" "not" "is" "on" "of"
 "we" "but" "have" "rates" "this" "as" "was"]
topic 9: ["loss" "vs" "revs" "shr" "year" "4th" "cts" "includes" "note" "inc" "lt"
 "qtr" "discontinued" "of" "dec" "operations" "dlr" "writedown" "ct"
 "losses"]
topic 10: ["fed" "customer" "says" "repurchase" "reserves" "federal" "agreements"
 "funds" "reserve" "repurchases" "sets" "temporary" "via" "system" "week"
 "add" "economists" "securities" "supply" "day"]
topic 11: ["oper" "excludes" "cts" "or" "discontinued" "dlrs" "gain" "note" "net"
 "operations" "of" "extraordinary" "year" "share" "exclude" "and" "vs"
 "shr" "tax" "from"]
topic 12: ["profit" "vs" "loss" "cts" "net" "1986" "nil" "tax" "year" "six" "4th"
 "revs" "ct" "pretax" "three" "shr" "group" "includes" "one" "two"]
topic 13: ["japan" "trade" "yen" "japanese" "to" "dollar" "dealers" "ec" "the"
 "tokyo" "bank" "surplus" "dollars" "tariffs" "against" "yeutter" "imports"
 "officials" "deficit" "and"]
topic 14: ["dlrs" "quarter" "share" "of" "earnings" "year" "1986" "1987" "first" "or"
 "in" "from" "for" "net" "and" "per" "gain" "includes" "company" "results"]
topic 15: ["cts" "qtly" "div" "record" "april" "pay" "prior" "dividend" "sets" "vs"
 "march" "quarterly" "payout" "lt" "may" "15" "10" "payable" "regular" "30"]
topic 16: ["oil" "crude" "prices" "barrel" "bbl" "opec" "50" "raises" "bpd" "gas"
 "barrels" "postings" "to" "price" "texas" "wti" "effective" "petroleum"
 "energy" "posted"]
topic 17: ["vs" "avg" "shrs" "net" "cts" "revs" "shr" "year" "mln" "4th" "lt"
 "diluted" "10" "31" "11" "12" "inc" "000" "13" "19"]
topic 18: ["it" "shares" "to" "said" "of" "stock" "its" "lt" "company" "inc" "and"
 "for" "common" "corp" "offer" "group" "split" "the" "share" "stake"]
topic 19: ["the" "of" "in" "and" "said" "to" "was" "by" "on" "for" "it" "will" "at"
 "with" "were" "which" "new" "is" "as" "from"]
```

Here is the code:

```
import numpy as np
import matplotlib.pyplot as plt
from nltk.corpus import reuters
from sklearn.feature_extraction import text

def objective(X, W, H):
    """ objective
    computes |X - WH|^2 for sparse X

    :type X: np.ndarray[m,n]
    :type W: np.ndarray[m,k]
    :type H: np.ndarray[k,n]
    """
    total = 0.
    cx = X.tocoo()
    for i,j,x in zip(cx.row, cx.col, cx.data):
        total += (x - np.inner(W[i],H[:,j]))**2
    return total

def nmf(X, k=20, n_iter=100, print_freq=5, verbose=False):
    """ nmf
    Non-negative matrix factorization on X using k latent
    factors. Algorithm from Lee and Seung 2001.
    """
    W = np.abs(np.random.randn(X.shape[0], k)*1e-3)
    H = np.abs(np.random.randn(k, X.shape[1])*1e-3)

    obj = [objective(X, W, H)]
    for i in range(n_iter):
        if i % print_freq == 0:
            print("[i={}] objective: {}".format(i, obj[-1]))
        H = H * (W.T @ X) / ((W.T @ W) @ H)
        W = W * (X @ H.T) / (W @ (H @ H.T))
        obj.append(objective(X, W, H))

    return W, H.T, obj

# RUN ON DATA #
X = np.array([
    " ".join(list(reuters.words(file_id))).lower()
    for file_id in reuters.fileids()
])
tfidf = text.TfidfVectorizer()
X = tfidf.fit_transform(X)

np.random.seed(0)
W, H, obj = nmf(X, k=20, n_iter=100)
plt.plot(obj)
plt.xlabel("iteration")
plt.ylabel("objective")
plt.title("Reuters NMF Convergence")
plt.savefig("nov_21/nmf_convergence.pdf")

# PRINT TOP WORDS FOR EACH TOPIC #
top_words = np.array(tfidf.get_feature_names())[
    np.argsort(H, axis=0)[::-1][:20]
].T # numpy wizardry
for i in range(top_words.shape[0]):
    print("topic {}: {}".format(i, top_words[i]))
```