Bayesian Clustering and Topic Discovery: Adventures with Gene Expression Data

Karren Dai Yang, Skanda Koppula {KARREN, SKOPPULA}@MIT.EDU
Massachusetts Institute of Technology, Cambridge, MA 02139 USA

1. Introduction

Tumors cell lines are composed of different subpopulations of cells which often exhibit shared patterns of gene expression. Biologists are interested in two key questions: given gene expressions values, (1) can we identify cell clusterings, and (2) can we identify clusters of biologically-related genes? Our goal with this project was to answer both these two questions using Bayesian methods.

In brief, we explored the use of three Bayesian clustering methods (a vanilla mixture, integrated topic-mixture, and non-parametric models) to address the first question, and two topic models (vanilla and a dynamic-topic LDA) to address the second.

1.1. Description of Data

Using a contemporary gene sequencing machine, we obtained samples of single-cell RNA-sequencing data (scRNA-seq) taken from tumors in mice. Our data consisted of the expression values of 22,712 genes for each of 4645 cells. In total, the dataset amounted to 0.86 GB, presenting significant computational challenges when attempting posterior estimation. Apart from a few computational tricks (online inference, multi-core parallelization), for most methods, we pruned non-informative genes from the dataset, ranking based on the deviation in value across cells [1]. The Seurat biological toolkit in R was also used for the purpose of low-variance feature selection (Satija, 2012). We eschewed dimensionality reduction techniques such as PCA because of loss of its direct feature interpretability. Prior researchers have labeled the cells in our dataset; there are a total of 9 cell categories. The complete dataset, as well our preprocessed and pruned versions, are openly available (Yang & Koppula).

[1] We recognize that this can bias towards noisy genes. We favor this method because it is simple and easy to implement, and a practice used in literature (Ling, 2012)

1.2. Prior work

Prior research has explored the use of various computational techniques to analyze gene expression data. Most commonly, Spearman and Pearson correlation metrics are frequently used infer sets of genes that cluster together (Xie, 2015; Borenszstein, 2017). Other techniques, including PCA followed by linear regression, has been used for expression-based cell clustering (Stegle, 2016). Yu et al. propose an unsupervised classifier ensemble as another approach to cell clustering (Yu, 2016).

More Bayesian approaches have also been tried in prior work from the Pe'er lab. (Prabhakaran, 2016) uses a Heirarchical Dirichlet Mixture Model to learn cell clusterings. (Azizi, 2017) builds on this to jointly learn optimal normalization pre-processing of the data. Bayesian networks have also been used in an attempt to learn gene dependencies from expression data (Pe'er, 2000). Our work uses different models to explore gene expression data, but where appropriate (e.g. in mixture models in Section 4), we compare results.

1.3. Structure of Report

We first discuss our experiments using Bayesian topic models to discover related sets of genes in a 'topic': LDA in Section 2 and Dynamic Time Models in Section 3. Then, we discuss our experiments in clustering: Dirichlet Mixtures in Section 4, Integrated Topic-Clustering in Section 5, and non-parametric models in Section 7. We conclude our paper with our observations from across all our studies.

For the purpose of reproducibility, all code can be found at `https://github.com/skoppula/882`.

2. Latent Dirichlet Allocation

2.1. Model Description

In the generative process for LDA, the topic assigned to each word is a drawn from the document's topic

distribution. The identity of the word is drawn from topic's word distribution. Assuming the reader is familiar with LDA, we relegate further details and formalization of the model to (Blei, 2003).

In the context of analyzing gene expression data, we are interested in discovering 'topics' that comprise of a set of top-N genes within the topic distribution that are biologically related. For example, together the genes may direct a specific chemical function in a cell. Biologists denote such sets of genes as 'gene modules' which can be cross-referenced with existing gene module databases.

2.2. Implementation

A first attempt using the built-in Python `lda` package resulting in early memory overflows during what we suspect was pre-allocation of per-document variables. The source code was not available, so we had few clues.

We switched to two open-source implementations: an online mean-field variational Bayes for posterior estimation (Nothman, 2017), and a broken `C++` Gibbs sampler for LDA (OpenDataGroup, 2015).

We fixed portions of the sampler to compile properly and extended the sampler to run across four cores. Details of our sampling procedure can be found in Appendix 9.1. We compared these two posterior estimation approaches using our entire dataset, using a 10% held-out testing partition. We experimented using $k = 5, 10, 25, 50$ topics. We did not require dataset pruning after these optimizations.

2.3. Experiments

In lieu of intrusion testing, we evaluated the interpretability of our topics (ranked lists of genes) using using hypergeometric tests (Wagner, 2015). In brief, this determines the probability that our sets of genes would be clustered together in accordance to with existing collections of genes catalogued by biologists in the gene module database `MSigDB` (Broad-Institute, 2015).

Figure 7 in the Appendix shows gene module matches in `MSigDB` for which the p-value of the match is at least less than 0.3. Notice that models with more topics tended to have more matches with higher significance. The p-values in our tests do *not* factor for multiple hypothesis corrections, so at the moment we are only using them as relative measures of model quality.

It is validating to see that many of the similar modules, such as the `BRCA1` and `DOXORUBICIN`, are cancer related, given that our expression assay was from tumor cells.

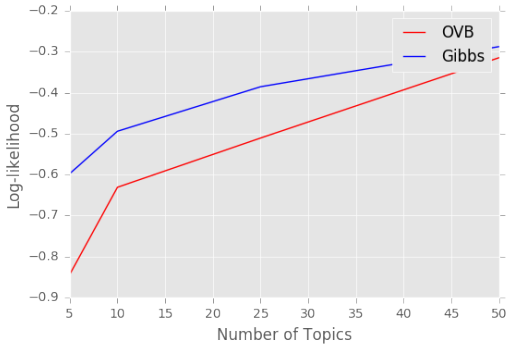

Figure 1. Log-likelihood of the held-out testing set across numbers of topics. Higher topic sizes are explored in later models.

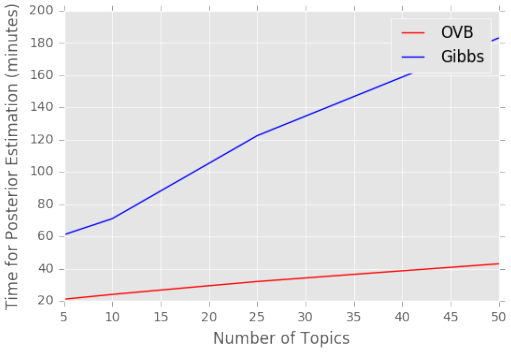

Figure 2. Comparison of the running times of each of the posterior estimation methods across various numbers of topics. Termination conditions across all runs were constant, and described in Appendix 9.1

Extracting the parameters of each model, we calculated log-likelihood on our held-out test set. This is shown in Figure 1. We found that for this dataset increasing the topic count increases log-likelihood, roughly linear from 5 to 50. Further experiments will need to test at what point log-likelihood drops off; this is one way to optimize the number of topics.

As an interesting aside, Figure 2 shows the time until inference completion (collected during our experiments). Gibbs appears to scale poorly with the parameter dimensionality, in contrast to online variational Bayes [2].

Posterior predictive checks to test the mutual information between the each cell and its words' topics is

[2]It is important to note that comparing the magnitude of the time to complete inference in Figure 2 is not particularly meaningful; as described in Appendix 9.1, we chose a reasonable guess for the number of iterations and optimization threshold after which to stop estimation

something that we did not have time, but would be interesting to examine. It's not clear that this independence assumption is true in the context of gene expression data, because of cross-talk and regulatory mechanisms between gene modules.

3. Dynamic Topic Model

3.1. Model Description

As tumor cells proliferate, they undergo a process called *differentiation*. The set of cells types during tumor emergence may differ from the set of types expressed days later. Correspondingly, patterns of gene expression may also change over time. The set of gene modules expressed in the cells, or the composition of each gene module (topic), may also change.

To capture these evolving topics, we look to dynamic topic models (Blei & Lafferty, 2006). In brief, dynamic topic models establish a conditional distribution over the hyperparameters α and β, that govern the document's topic distribution and topic's word distribution, respectively. The distribution over each hyperparameter is conditioned on the prior in the previous time step, allowing changes to topic composition and assignment. This can be seen graphically in the plate model in Figure 3. For convenience, we provide a concise description of the generative process for the implemented heirarchical model for a time slice t in Appendix 9.2.

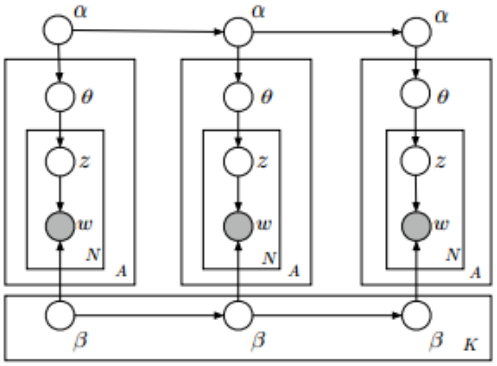

Figure 3. Plate diagram for the Dynamic Topic Model. Reproduced from (Blei & Lafferty, 2006).

Note that in contrast to LDA, dynamic topic models use a logistic-normal to express proportion uncertainty. This is stated more explicitly in Appendix 9.2.

3.2. Implementation

Unfortunately, due to the non-conjugacy of the Gaussian/Multinomial in our logistic-normal setup, integrating out parameters for any sort of Gibbs sampling becomes hard. Instead, the original DTM paper uses a variational approximation based on a Kalman filter, that preserves time dependencies, unlike a mean-field approximation.

Re-implementing the variational approximation, while educational and interesting, would be a project of itself, so our first attempt in employing dynamic time models to our gene expression data re-used the Kalman Filter-based variational inference used by the original authors, published recently by the Blei Lab (Blei & Gerrish, 2015). The results we show in the subsequent section are using a wrapper we've implemented around the group's inference code.

The authors stumbled upon a recent arXiv pre-print that proposed a set of sampler update rules to create a correct DTM Gibbs sampler (Bhadury, 2016). Using the probabalistic programming framework `Edward`, and modifying the in-built sampler to follow these updates, we were able to obtain a working Gibbs-sampler for sample time-sliced data for a dynamic topic model defined in Edward (Tran et al., 2016). While we don't have time to conduct benchmarks or re-run our results right now, after verifying that results are consistent, the authors hope to submit a merge for this Edward sketch into the mainstream Edward examples library this coming summer.

3.3. Experiments

We subdivided our data into three seperate time slices, and pruned each slice, retaining the top 20% of high-variance genes. We experimented using $k = 15$, 30, and 50 topics.

Similar to our analysis in Section 2, we used the hypergeometric test to evaluate interpretability.

Figure 8 in the Appendix shows gene module matches for which the p-value is less than 0.3, across the three time slices and the three topic count sizes. Interestingly, we notice a very different set of matched gene modules, with exception of the `KLF1` module[3]. Like before, we notice that some of the identified topics are relevant to tumor cells: `ST_TUMOR_NECROSIS_FACTOR_PATHWAY` and `WEST_ADRENOCORTICAL_TUMOR_MARKERS_UP` to name the two most prominent topics.

We do not see that much variation across time-slices, demonstrating that our time-slice partitions are largely uniform. For example, for model $k = 50$, we see that topic 20 consistently matches with MSigDB module `PILON_KLF1_TARGETS_UP`, suggesting that its gene

[3]This could very well be a result of our dataset pruning, which we did to keep inference runtimes manageable

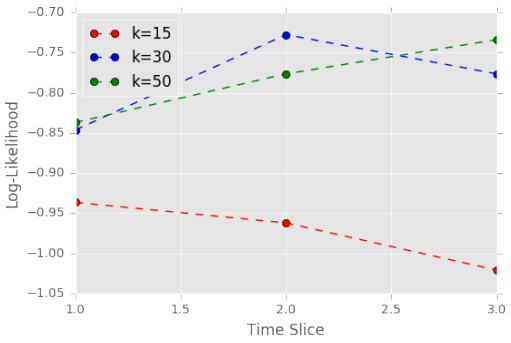

Figure 4. Log-likelihood of the held-out testing set across our time slices for varying numbers of topics.

composition is not varying.

Models with more topics tended to have more matches with higher significance. But again, the p-values in our tests do *not* factor for multiple hypothesis corrections, so at the moment we are only using them as relative measures of model quality.

After extracting the mode of learned parameters, we evaluated the log-likelihood of our held-out test set, also partitioned by time. This is shown in Figure 4. We find a leveling of log-likelihood after $k = 30$.

With more time, we would ideally repeat the experiments to obtain confidence intervals on these experiments, and examine the gene-module/cell/time-slice mutual information.

4. Mixture Model

4.1. Model Description

The second question – whether we could learn cell groupings – we tackled using a canonical Gaussian mixture model. In brief, the model assigns every data point to a cluster; every cluster is Gaussian whose parameters are learned through posterior inference. A plate diagram can be found at (Ben Wing, 2016). This model is different from the previously mentioned Pe'er papers which use Dirichlet Processes Mixtures to study expression data. By plotting histograms of whitened gene expression values, we've found that for many genes the distribution is Gaussian across our data points. This hints that the generative process in a Gaussian mixture model is plausible.

4.2. Implementation

We implemented the mixture model in **Edward**, the probabalistic programming framework on top of Tensorflow. This allowed us to experiment with different samplers and variational inference approximations.

The results we show below are for our runs using a Gibbs sampler, with a fixed burn-in of number of samples (200), a fixed number of sampling iterations (500).

4.3. Experiments

Figure 6 shows an estimated cluster assignment for one-hundred sampled data point. We see that while our learned assignments do capture some of the clustering, there is significant bias toward one group (the burgundy cluster).

Following this trend, we examined the set of unique clusters assigned in the posterior. The results are shown in Table 4.3. Interestingly, the model consistently uses less clusters than is allocated in the model. We are unsure why this is the case.

Finally, we examine the log-likelihood of a held-out test set three different runs of posterior estimation with our sampler. This is shown in Figure 5. The runs are consistent with each other, and the likelihood is largely invariant across the five different cluster counts we tried, $k = \{10, 25, 50, 75, 50\}$. There are an estimated 9 true cell types in the original dataset, which could explain this result.

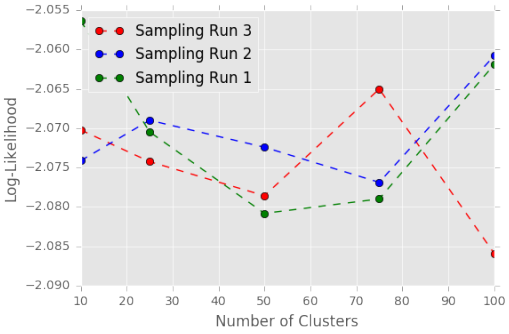

Figure 5. Log-likelihood of held-out test set, on three seperate runs of Gibbs sampling procedure, described above

# of Clusters	# of Clusters Used
10	3
25	9
50	16
75	18
100	20

Table 1. The number of clusters defined in the model appears to be more than the actual number of unique clusters assigned to the data points after posterior estimation. The estimated true number of categories is 9.

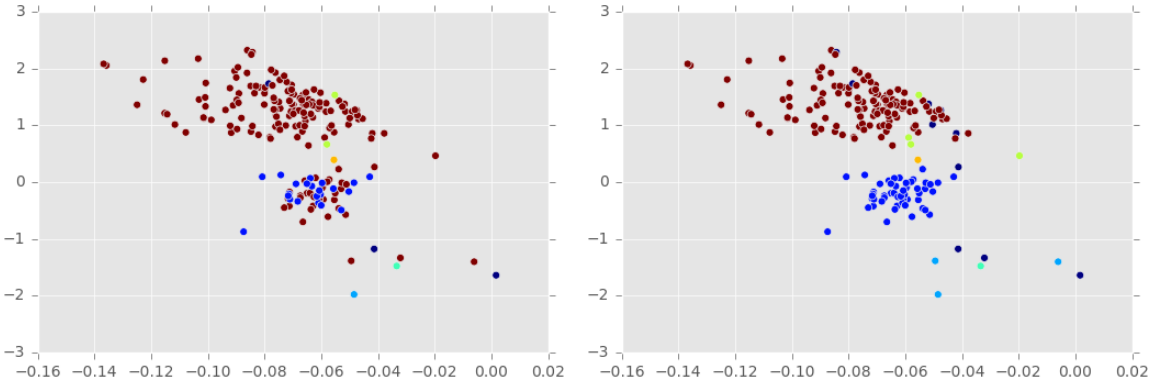

Figure 6. 2D-projection of sampled gene expression data points, colored by predicted cluster (left) and by true cluster assignments (right). Axis are the two learned PCA dimensions.

5. Integrated Topic-Clustering Model

5.1. Model Description

5.2. Implementation

5.3. Experiments

6. Non-parametric Models: IBP and HDP

6.1. Model Description

6.2. Implementation

6.3. Experiments

7. Conclusions

8. Breakdown of Work

The implementation(s) and experiments involving LDA, Gaussian Mixtures, and Dynamic Time Model was completed by SK (Section 2, Section 4, and Section 3). KY completed the implementations and experiments involved Integrated Topic-Clustering, non-parametric models, (Section 5 and Section 7) and the shared gene enrichment test implementation. The authors contributed equally in this work.

References

Azizi, et al. Bayesian inference for single-cell clustering and imputing. https://genomicscomputbiol.org/ojs/index.php/GCB/article/view/46/35, Genomics 2017.

Ben Wing, other Wikipedia contributors. Bayesian gaussian mixture model. https://en.wikipedia.org/wiki/Mixture_model#/media/File:Bayesian-gaussian-mixture.svg, 2016.

Bhadury, et al. Scaling up dynamic topic models. https://arxiv.org/abs/1602.06049, IWSS 2016.

Blei, David M. and Gerrish, Sean. Dynamic topic models and the document influence model. https://github.com/blei-lab/dtm, GitHub 2015.

Blei, David M. and Lafferty, John D. Dynamic topic models. https://mimno.infosci.cornell.edu/info6150/readings/dynamic_topic_models.pdf, ICML 2006.

Blei, et al. Latent dirichlet allocation. http://www.jmlr.org/papers/volume3/blei03a/blei03a.pdf, JMLR 2003.

Borenszstein, et al. Xist-dependent imprinted x inactivation and the early developmental consequences of its failure. http://www.nature.com/nsmb/journal/v24/n3/fig_tab/nsmb.3365_SF1.html, Nature Structural & Molecular Biology 2017.

Broad-Institute. Molecular signatures database v6.0. http://software.broadinstitute.org/gsea/msigdb, PeerJ 2015.

Hoffman, et al. Online learning for latent dirichlet allocation.

Ling, et al. Improving relative-entropy pruning using statistical significance. https://www.cs.cmu.edu/~awb/papers/coling2012/rep_coling2012.pdf, ACL 2012.

Nothman, Joel. Online latent dirichlet allocation with variational inference. https://github.com/scikit-learn/scikit-learn/blob/master/sklearn/decomposition/online_lda.py, 2017.

OpenDataGroup. C++ implementation of latent dirichlet allocation. https://github.com/openbigdatagroup/plda/blob/master/lda.cc, 2015.

Pe'er, et al. Using bayesian networks to analyze expression data. http://www.cs.huji.ac.il/~nir/Papers/FLNP1Full.pdf, Genomics 2000.

Prabhakaran, et al. Dirichlet process mixture model for correcting technical variation in single-cell gene expression data. http://www.c2b2.columbia.edu/danapeerlab/html/pub/prabhakaran16.pdf, ICML 2016.

Satija, et al. Spatial reconstruction of single-cell gene expression data. http://www.nature.com/nbt/journal/v33/n5/full/nbt.3192.html, Nature Biotechnology 2012.

Stegle, et al. Computational and analytical challenges in single-cell transcriptomics. https://www.nature.com/nrg/journal/v16/n3/full/nrg3833.html, Nature Methods 2016.

Tran, Dustin, Kucukelbir, Alp, Dieng, Adji B., Rudolph, Maja, Liang, Dawen, and Blei, David M. Edward: A library for probabilistic modeling, inference, and criticism. *arXiv preprint arXiv:1610.09787*, 2016.

Wagner, Florian. The xl-mhg test for enrichment: A technical report. https://arxiv.org/pdf/1507.07905.pdf, PeerJ 2015.

Xie, et al. SINCERA: A pipeline for single-cell rna-seq profiling analysis. http://months.plos.org/ploscompbiol/article?id=10.1371/month.pcbi.1004575, PLoS 2015.

Yang, Karren and Koppula, Skanda. Tumor cell melanoma data. https://www.dropbox.com/sh/tdrplhiu3mzvmeu/AADFlZITyPX_-I4O7jDuruJra?dl=0.

Yu, et al. SC3 - consensus clustering of single-cell rna-seq data. http://biorxiv.org/content/early/2016/09/02/036558, Nature Methods 2016.

9. Appendix

9.1. Latent Dirichlet Allocation

This section includes Figure 7 on the next page.

For our Gibb's sampler, we had a fixed burn-in of number of samples (200), a fixed number of sampling iterations after that (500). We didn't extensively explore varying these values, but trying out significantly more iterations (700) didn't seem to change the topic's word distributions significantly. There was one sampling chain on each of four cores.

9.2. Dynamic Topic Model

This section includes Figure 8 on the next page.

Here is a description of the generative process for our dynamic topic model:

1. Draw a new topic composition hyperparameter $\beta \leftarrow \mathbb{N}(\beta_{t-1}, \sigma I^2)$
2. Draw a new document composition hyperparameter $\alpha \leftarrow \mathbb{N}(\alpha_{t-1}, \delta I^2)$
3. For each document:
   (a) Draw a new document topic distribution $\theta \leftarrow \pi(\mathbb{N}(\alpha_t))$
   (b) For every word in the document:
      i. Draw the word topic assignment $Z \leftarrow \text{Mult}(\theta)$
      ii. Draw the word identity $Z \leftarrow \text{Mult}(\pi(\beta_{t,z}))$

Here, $\pi(x_i)$ is the softmax function $\frac{exp(x_i)}{\sum_k exp(x_k)}$ (Blei & Lafferty, 2006).

```
GIBBS TOPICS
        cluster_size: 5
        cluster_size: 10
                match: topic  1 ,  KINSEY_TARGETS_OF_EWSR1_FLII_FUSION_UP , pval: 0.041958041958
                match: topic  1 ,  PILON_KLF1_TARGETS_DN , pval: 0.137062937063
                match: topic  6 ,  GRAESSMANN_APOPTOSIS_BY_DOXORUBICIN_DN , pval: 0.263736263736
        cluster_size: 25
                match: topic  8 ,  PILON_KLF1_TARGETS_DN , pval: 0.263736263736
                match: topic  22 ,  KINSEY_TARGETS_OF_EWSR1_FLII_FUSION_UP , pval: 0.041958041958
                match: topic  22 ,  PILON_KLF1_TARGETS_DN , pval: 0.137062937063
                match: topic  22 ,  GRAESSMANN_APOPTOSIS_BY_DOXORUBICIN_DN , pval: 0.263736263736
        cluster_size: 50
                match: topic  2 ,  ROME_INSULIN_TARGETS_IN_MUSCLE_UP , pval: 0.193006993007
                match: topic  3 ,  PILON_KLF1_TARGETS_DN , pval: 0.00699300699301
                match: topic  6 ,  PILON_KLF1_TARGETS_DN , pval: 0.263736263736
                match: topic  7 ,  BLALOCK_ALZHEIMERS_DISEASE_UP , pval: 0.018648018648
                match: topic  17 ,  KINSEY_TARGETS_OF_EWSR1_FLII_FUSION_UP , pval: 0.041958041958
                match: topic  17 ,  PILON_KLF1_TARGETS_DN , pval: 0.193006993007
                match: topic  22 ,  PILON_KLF1_TARGETS_DN , pval: 0.153846153846
                match: topic  24 ,  WEI_MYCN_TARGETS_WITH_E_BOX , pval: 0.0839160839161
OVB TOPICS
        cluster_size: 5
        cluster_size: 10
                match: topic  5 ,  PILON_KLF1_TARGETS_DN , pval: 0.263736263736
        cluster_size: 25
                match: topic  5 ,  DIAZ_CHRONIC_MEYLOGENOUS_LEUKEMIA_UP , pval: 0.263736263736
                match: topic  8 ,  KINSEY_TARGETS_OF_EWSR1_FLII_FUSION_UP , pval: 0.041958041958
                match: topic  15 ,  PUJANA_ATM_PCC_NETWORK , pval: 0.263736263736
                match: topic  15 ,  PUJANA_BRCA1_PCC_NETWORK , pval: 0.263736263736
        cluster_size: 50
                match: topic  6 ,  GRAESSMANN_APOPTOSIS_BY_DOXORUBICIN_DN , pval: 0.263736263736
                match: topic  15 ,  PUJANA_ATM_PCC_NETWORK , pval: 0.263736263736
                match: topic  22 ,  MARSON_BOUND_BY_FOXP3_UNSTIMULATED , pval: 0.263736263736
                match: topic  28 ,  PILON_KLF1_TARGETS_DN , pval: 0.201398601399
                match: topic  38 ,  PUJANA_BRCA1_PCC_NETWORK , pval: 0.263736263736
                match: topic  41 ,  MARSON_BOUND_BY_FOXP3_UNSTIMULATED , pval: 0.263736263736
                match: topic  42 ,  KINSEY_TARGETS_OF_EWSR1_FLII_FUSION_UP , pval: 0.193006993007
```

Figure 7. Matches between the gene collections found in LDA topics and published gene sets in `MSigDB`. 'Cluster size' refers to the number of topics in the model

```
k=15
        t=1
                match: topic 10 , PILON_KLF1_TARGETS_UP , pval: 0.05484091166
                match: topic 11 , ST_TUMOR_NECROSIS_FACTOR_PATHWAY , pval: 0.13159861000
        t=2
        t=3
                match: topic 2 , ST_INTEGRIN_SIGNALING_PATHWAY , pval: 0.22324501174
                match: topic 10 , PILON_KLF1_TARGETS_UP , pval: 0.25017876663

k=30
        t=1
                match: topic 26 , WEST_ADRENOCORTICAL_TUMOR_MARKERS_UP , pval: 0.04716140093
                match: topic 27 , ST_TUMOR_NECROSIS_FACTOR_PATHWAY , pval: 0.15884551563
        t=2
                match: topic 12 , KEGG_PARKINSONS_DISEASE , pval: 0.19929937855
                match: topic 26 , WEST_ADRENOCORTICAL_TUMOR_MARKERS_UP , pval: 0.08559859059
        t=3
                match: topic 11 , TIMOFEEVA_GROWTH_STRESS_VIA_STAT1_DN , pval: 0.08405900408
                match: topic 27 , ST_TUMOR_NECROSIS_FACTOR_PATHWAY , pval: 0.09390276030

k=50
        t=1
                match: topic 13 , WEST_ADRENOCORTICAL_TUMOR_MARKERS_UP , pval: 0.00922042410
                match: topic 20 , HOLLMANN_APOPTOSIS_VIA_CD40_DN , pval: 0.05797811105
                match: topic 20 , PILON_KLF1_TARGETS_UP , pval: 0.05483747800
                match: topic 20 , WEST_ADRENOCORTICAL_TUMOR_MARKERS_UP , pval: 0.02003797479
        t=2
                match: topic 15 , ST_P38_MAPK_PATHWAY , pval: 0.19462717810
        t=3
                match: topic 7 , ST_TUMOR_NECROSIS_FACTOR_PATHWAY , pval: 0.20037158939
                match: topic 13 , WEST_ADRENOCORTICAL_TUMOR_MARKERS_UP , pval: 0.05761453734
                match: topic 20 , PILON_KLF1_TARGETS_UP , pval: 0.09429082710
```

Figure 8. Matches between the gene collections found using a dynamic topic model and published gene sets in `MSigDB`. k refers to the number of topics in the model. t refers to the time slice in the model.