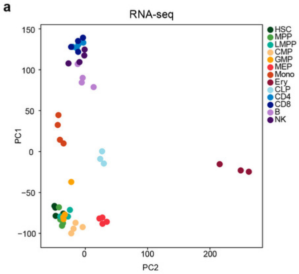
PLIER:Pathway-Level Information Extractor

Maria Chikina
Computational and Systems Biology
University of Pittsburgh

Principal component analysis of gene expression data: how to interpret?

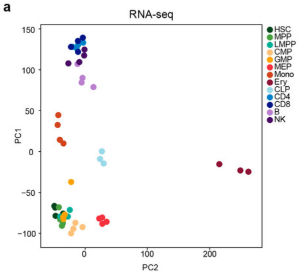
Principal component analysis of gene expression data: how to interpret?

- Interpretation 1: A plot of the first 2-3 principal components (PCs) should separate samples according to the phenotype of interest.



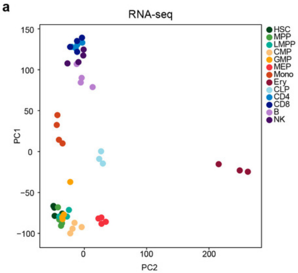
Principal component analysis of gene expression data: how to interpret?

- Interpretation 1: A plot of the first 2-3 principal components (PCs) should separate samples according to the phenotype of interest.
  - Controlled model system experiments.
  - Phenotypes with large effects, such as tissues or cancer subtypes.



Principal component analysis of gene expression data: how to interpret?

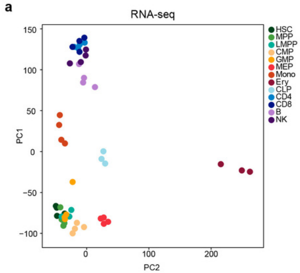
- Interpretation 1: A plot of the first 2-3 principal components (PCs) should separate samples according to the phenotype of interest.
  - Controlled model system experiments.
  - Phenotypes with large effects, such as tissues or cancer subtypes.



- Interpretation 2: The first few PCs are nuisance variables that should be removed from the data.

Principal component analysis of gene expression data: how to interpret?

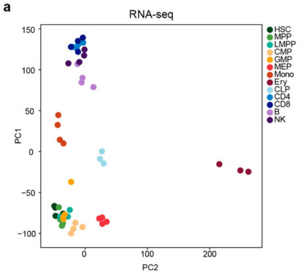
- Interpretation 1: A plot of the first 2-3 principal components (PCs) should separate samples according to the phenotype of interest.
    - Controlled model system experiments.
    - Phenotypes with large effects, such as tissues or cancer subtypes.

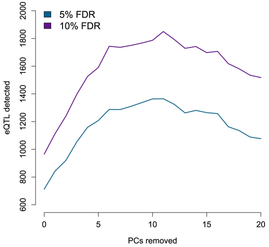


- Interpretation 2: The first few PCs are nuisance variables that should be removed from the data.
    - eQTL discovery: removing PCs improves results.

Principal component analysis of gene expression data: how to interpret?

- Interpretation 1: A plot of the first 2-3 principal components (PCs) should separate samples according to the phenotype of interest.
  - Controlled model system experiments.
  - Phenotypes with large effects, such as tissues or cancer subtypes.
- Interpretation 2: The first few PCs are nuisance variables that should be removed from the data.
  - eQTL discovery: removing PCs improves results.

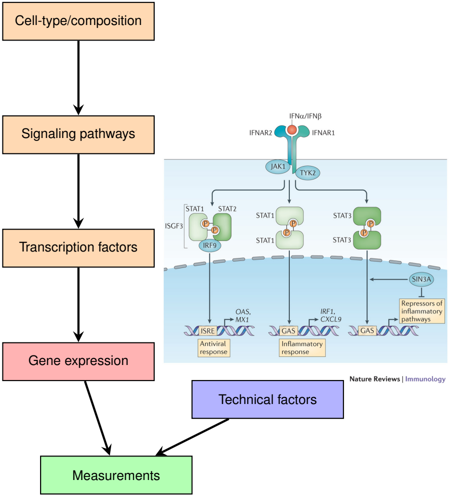




Generative model for gene expression data

- Gene expression is driven by **upstream factors** that give rise to the observed data structure.

- PCA gives us a representation of these **upstream factors** but not a one-to-one correspondence.



Nature Reviews | Immunology

General matrix decompositions applied to gene expression

General matrix decompositions applied to gene expression

- Low rank matrix approximations (such as ones given by PCA) are effective because a limited number of **upstream factors** explain a large fraction of measurement variance.

General matrix decompositions applied to gene expression

- Low rank matrix approximations (such as ones given by PCA) are effective because a limited number of **upstream factors** explain a large fraction of measurement variance.
  Given gene expression matrix $Y_{g \times s}$:

$$\textbf{MINIMIZE} \quad ||Y_{g \times s} - Z_{g \times k} B_{k \times s}||_F^2$$

General matrix decompositions applied to gene expression

- Low rank matrix approximations (such as ones given by PCA) are effective because a limited number of **upstream factors** explain a large fraction of measurement variance.
  Given gene expression matrix $Y_{g \times s}$:

$$\textbf{MINIMIZE} \quad ||Y_{g \times s} - Z_{g \times k} B_{k \times s}||_F^2$$

- *B* contains the principal components (PCs) or more generally latent variables (LVs). *Z* contains the “loadings” (effect of each LV on the genes).

General matrix decompositions applied to gene expression

- Low rank matrix approximations (such as ones given by PCA) are effective because a limited number of **upstream factors** explain a large fraction of measurement variance.
  Given gene expression matrix $Y_{g\times s}$:

$$\textbf{MINIMIZE} \quad ||Y_{g\times s} - Z_{g\times k}B_{k\times s}||_F^2$$

- B contains the principal components (PCs) or more generally latent variables (LVs). Z contains the “loadings” (effect of each LV on the genes).
- We hope that the individual vectors B_i (latent variables) are meaningful. SVD (PCA) only guarantees minimum error—it doesn’t guarantee anything about the interpretability of B.

General matrix decompositions applied to gene expression

- Low rank matrix approximations (such as ones given by PCA) are effective because a limited number of **upstream factors** explain a large fraction of measurement variance.
  Given gene expression matrix $Y_{g \times s}$:

  $$\textbf{MINIMIZE} \quad ||Y_{g \times s} - Z_{g \times k} B_{k \times s}||_F^2$$

- B contains the principal components (PCs) or more generally latent variables (LVs). Z contains the “loadings” (effect of each LV on the genes).
- We hope that the individual vectors B_i (latent variables) are meaningful. SVD (PCA) only guarantees minimum error—it doesn't guarantee anything about the interpretability of B.
- Other methods that constrain Z to be sparse or positive may recover more meaningful structure.

General matrix decompositions applied to gene expression

- Low rank matrix approximations (such as ones given by PCA) are effective because a limited number of **upstream factors** explain a large fraction of measurement variance.
  Given gene expression matrix $Y_{g \times s}$:

$$\textbf{MINIMIZE} \quad ||Y_{g\times s} - Z_{g\times k} B_{k\times s}||_F^2$$

- B contains the principal components (PCs) or more generally latent variables (LVs). Z contains the “loadings” (effect of each LV on the genes).
- We hope that the individual vectors B_i (latent variables) are meaningful. SVD (PCA) only guarantees minimum error—it doesn’t guarantee anything about the interpretability of B.
- Other methods that constrain Z to be sparse or positive may recover more meaningful structure.

$$\begin{aligned} \textbf{MINIMIZE} \quad & ||Y - ZB||_F^2 + \lambda ||Z||_{L^1} \\ \textbf{SUBJECT TO} \quad & Z > 0. \end{aligned}$$

Can we recover the data generating process from general matrix decompositions?

Can we recover the data generating process from general matrix decompositions?

- We construct an example with 7 **upstream factors**; can we recover them?

Can we recover the data generating process from general matrix decompositions?

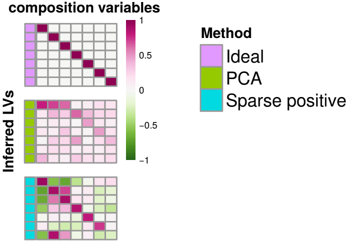
- We construct an example with 7 **upstream factors**; can we recover them?
- We can make this problem quite hard by making some upstream factors have low variance

Can we recover the data generating process from general matrix decompositions?

- We construct an example with 7 **upstream factors**; can we recover them?
- We can make this problem quite hard by making some upstream factors have low variance (very realistic: e.g., some cell-types have low abundance).

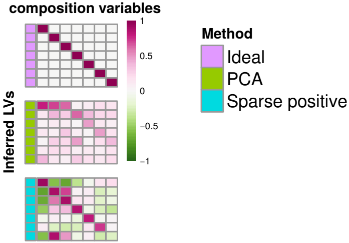
Can we recover the data generating process from general matrix decompositions?

- We construct an example with 7 **upstream factors**; can we recover them?
- We can make this problem quite hard by making some upstream factors have low variance (very realistic: e.g., some cell-types have low abundance).



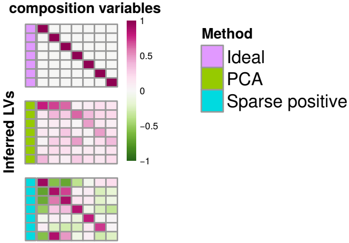
Can we recover the data generating process from general matrix decompositions?

- We construct an example with 7 **upstream factors**; can we recover them?
- We can make this problem quite hard by making some upstream factors have low variance (very realistic: e.g., some cell-types have low abundance).
- PCA is very restrictive: each component is orthogonal.



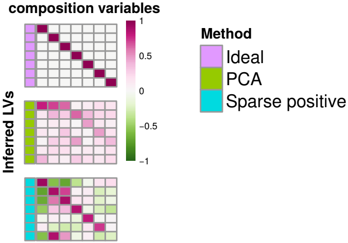
Can we recover the data generating process from general matrix decompositions?

- We construct an example with 7 **upstream factors**; can we recover them?
- We can make this problem quite hard by making some upstream factors have low variance (very realistic: e.g., some cell-types have low abundance).
- PCA is very restrictive: each component is orthogonal.
- If we constrain the decomposition to have sparse and positive loadings we can recover some, but not all, variables of interest.



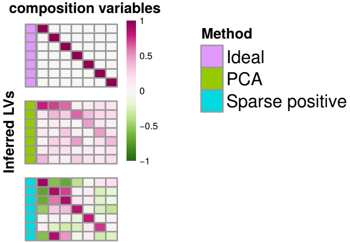
Can we recover the data generating process from general matrix decompositions?

- We construct an example with 7 **upstream factors**; can we recover them?
- We can make this problem quite hard by making some upstream factors have low variance (very realistic: e.g., some cell-types have low abundance).
- PCA is very restrictive: each component is orthogonal.
- If we constrain the decomposition to have sparse and positive loadings we can recover some, but not all, variables of interest.
  - These methods are data agnostic, they don't make use of gene identities!



Can we recover the data generating process from general matrix decompositions?

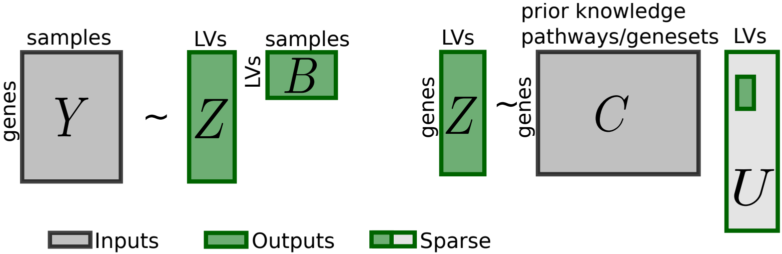
- We construct an example with 7 **upstream factors**; can we recover them?
- We can make this problem quite hard by making some upstream factors have low variance (very realistic: e.g., some cell-types have low abundance).
- PCA is very restrictive: each component is orthogonal.
- If we constrain the decomposition to have sparse and positive loadings we can recover some, but not all, variables of interest.
  - These methods are data agnostic, they don't make use of gene identities!
  - We want not just the most parsimonious but also the most biologically meaningful decomposition.



PLIER: Pathway-Level Information ExtractoR

Idea: Make use of gene identities.

SUBJECT TO rank(Z) $= k$, rank(B) $= k$, $U > 0$, $Z > 0$.



Prior knowledge matrix C is a binary geneset representation, where each column is a potentially co-regulated set of genes. Number of genesets is many times larger than k.

Implementation Details

Implementation Details

- Non-convex optimization problem is solved by block-coordinate minimization

Implementation Details

- Non-convex optimization problem is solved by block-coordinate minimization

- All constants are set automatically

Implementation Details

- Non-convex optimization problem is solved by block-coordinate minimization

- All constants are set automatically

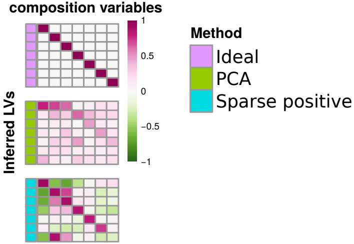
- Running time depends on the size of the data and size of C

Implementation Details

- Non-convex optimization problem is solved by block-coordinate minimization
- All constants are set automatically
- Running time depends on the size of the data and size of C
- We pre-compute the inverse of C and use it to find a set of active genesets in each iteration to be optimized with the elastic-net penalty

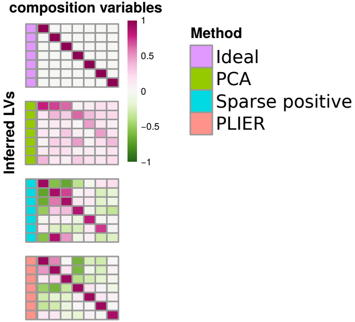
Recovering the pathway effects with PLIER

Revisit the toy example



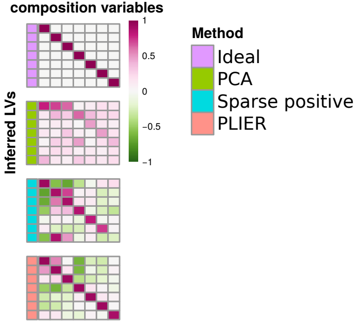
Recovering the pathway effects with PLIER

Revisit the toy example

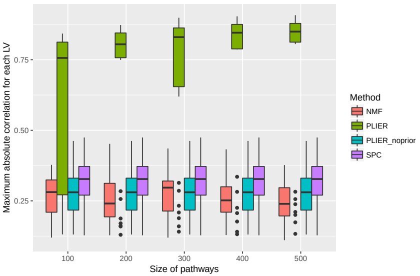


Recovering the pathway effects with PLIER

Revisit the toy example

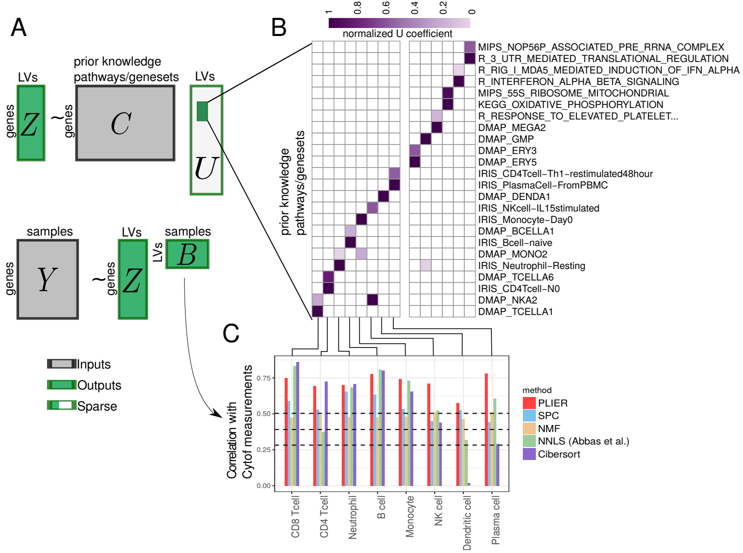


- Performance across repeated simulations
- Recovering 30 pathway effects from a prior information database of 1000 pathways



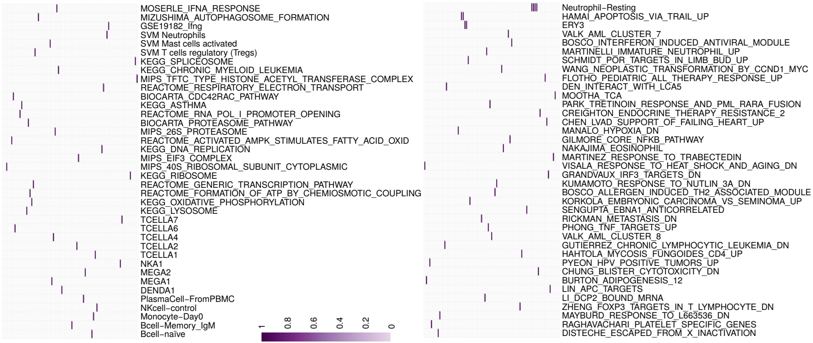
How do we use PLIER?

Example on real human blood dataset (35 samples) with directly measured by Cytof

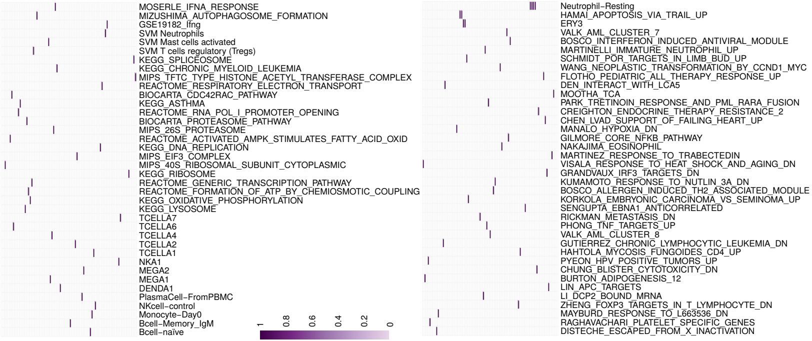


U matrix for a large dataset (DGN)

U matrix for a large dataset (DGN)

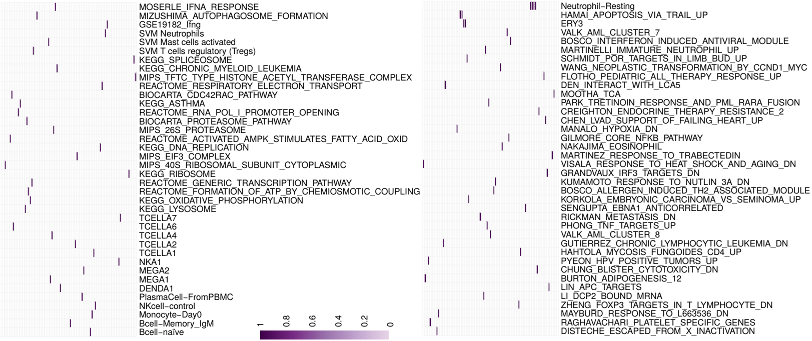


U matrix for a large dataset (DGN)



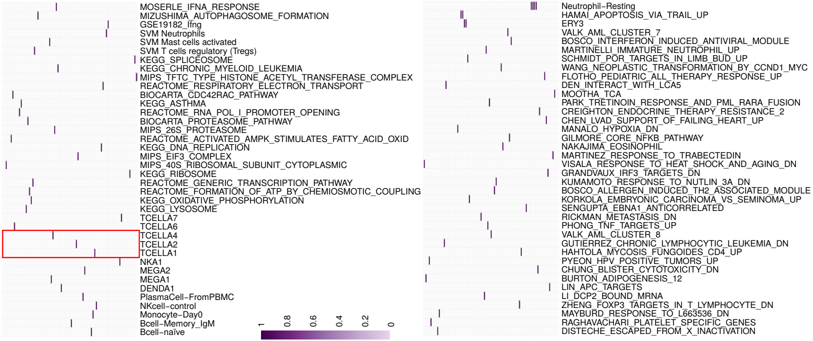
- How do we know the pathways are real? we zero-out a random 1/5 of the genes for every pathway before optimization and check if we get them back in the loading.

U matrix for a large dataset (DGN)



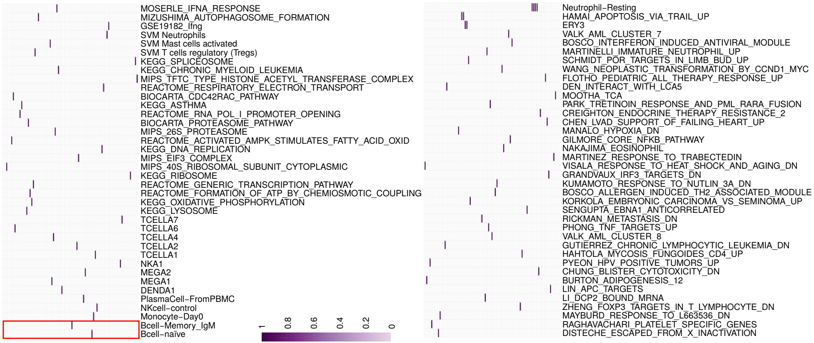
- How do we know the pathways are real? we zero-out a random 1/5 of the genes for every pathway before optimization and check if we get them back in the loading.
- We can see many cell types.

U matrix for a large dataset (DGN)



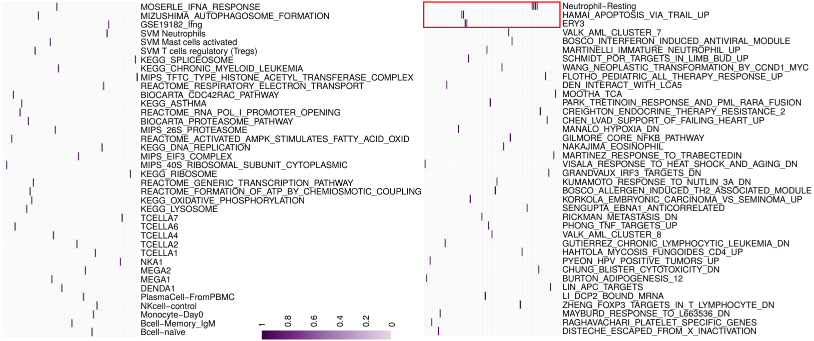
- How do we know the pathways are real? we zero-out a random 1/5 of the genes for every pathway before optimization and check if we get them back in the loading.
- We can see many cell types.
    - 3 kinds of CD8 T cells.

U matrix for a large dataset (DGN)



- How do we know the pathways are real? we zero-out a random 1/5 of the genes for every pathway before optimization and check if we get them back in the loading.
- We can see many cell types.
  - 3 kinds of CD8 T cells.
  - Naive and memory B-cells.

U matrix for a large dataset (DGN)



- How do we know the pathways are real? we zero-out a random 1/5 of the genes for every pathway before optimization and check if we get them back in the loading.
- We can see many cell types.
  - 3 kinds of CD8 T cells.
  - Naive and memory B-cells.
  - Very high frequency cell-types have multiple LVs.

How do we use PLIER?

PLIER latent variables can be plugged into any downstream analysis that would normally be done at the gene level–for example eQTLs.

LV id	LV name	snps	cis-Gene(s)	corrected p-value
44	Mega/platelet 1	rs1354034	ARHGEF3	$< 1.45e-10$
133	Mega/platelet 2	rs1354034	ARHGEF3	0.01547
120	Histones	rs1354034	ARHGEF3	0.01889
97	Zinc fingers, pseudogenes	rs1471738	SENP7	$< 1.45e-10$
56	PLAGL1 associated, myeloid	rs9321957	PLAGL1	3.6e-05
42*	IKZF1 associated, myeloid	rs10251980	IKZF1	$< 1.45e-10$
17	NEK6 associated, myeloid	rs16927294	NEK6	0.00360
67	Neutrophils	rs13289095	PKN3,SET,ZDHHC12	0.01888
55*	NFE2 associated, erythrocyte	rs35979828	NFE2	$< 1.45e-10$
21	Interferon-gamma	rs3184504	SH2B3	5.9e-05
40	NFKB/TNF	rs12100841	PPP2R3C	0.00204
16	Myeloid/ILC	rs1138358	BCL2A1,MTHFS,ST20	0.00025

Interferon-gamma LV21 uses 3 pathways:

- REACTOME_INTERFERON_GAMMA_SIGNALING
- GSE19182 Ifng
- SANA_RESPONSE_TO_IFNG_UP

How do we use PLIER?

PLIER latent variables can be plugged into any downstream analysis that would normally be done at the gene level–for example eQTLs.

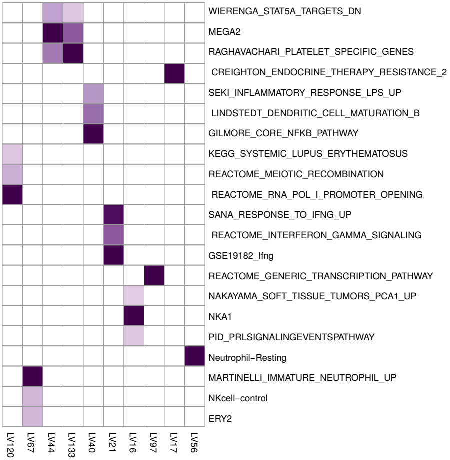
LV id	LV name	snps	cis-Gene(s)	corrected p-value
44	Mega/platelet 1	rs1354034	ARHGEF3	< 1.45e-10
133	Mega/platelet 2	rs1354034	ARHGEF3	0.01547
120	Histones	rs1354034	ARHGEF3	0.01889
97	Zinc fingers, pseudogenes	rs1471738	SENP7	< 1.45e-10
56	PLAGL1 associated, myeloid	rs9321957	PLAGL1	3.6e-05
42*	IKZF1 associated, myeloid	rs10251980	IKZF1	< 1.45e-10
17	NEK6 associated, myeloid	rs16927294	NEK6	0.00360
67	Neutrophils	rs13289095	PKN3,SET,ZDHHC12	0.01888
55*	NFE2 associated, erythrocyte	rs35979828	NFE2	< 1.45e-10
21	Interferon-gamma	rs3184504	SH2B3	5.9e-05
40	NFKB/TNF	rs12100841	PPP2R3C	0.00204
16	Myeloid/ILC	rs1138358	BCL2A1,MTHFS,ST20	0.00025

Interferon-gamma LV21 uses 3 pathways:

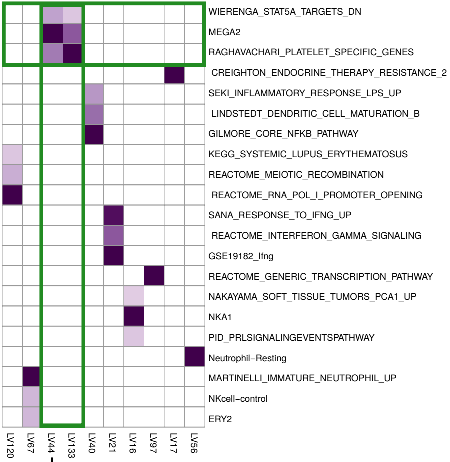
- REACTOME_INTERFERON_GAMMA_SIGNALING
- GSE19182 Ifng
- SANA_RESPONSE_TO_IFNG_UP

A single locus controls 2 pathway effects

LV eQTLs pathway associations



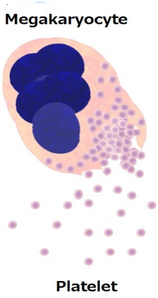
A single locus controls 2 pathway effects



LV eQTLs pathway associations

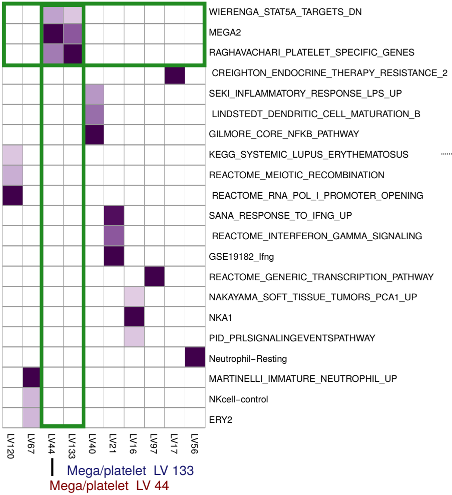
Mega/platelet LV 133

Mega/platelet LV 44

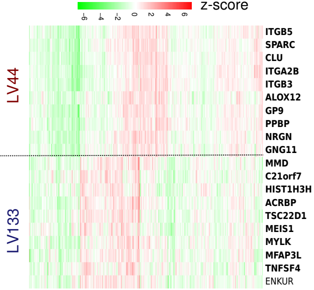


A single locus controls 2 pathway effects

LV eQTLs pathway associations

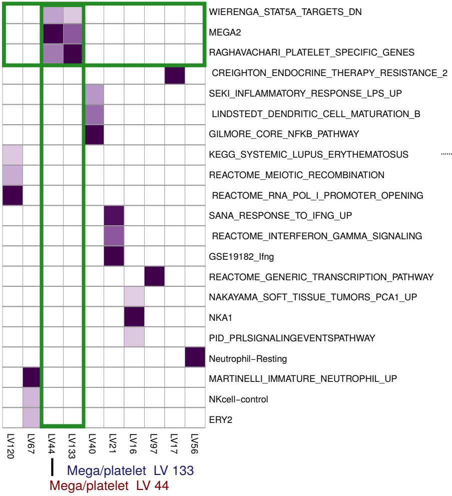


Top genes for mega/platelet LVs

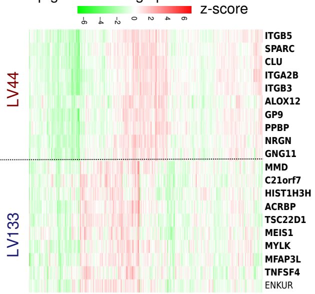


A single locus controls 2 pathway effects

LV eQTLs pathway associations

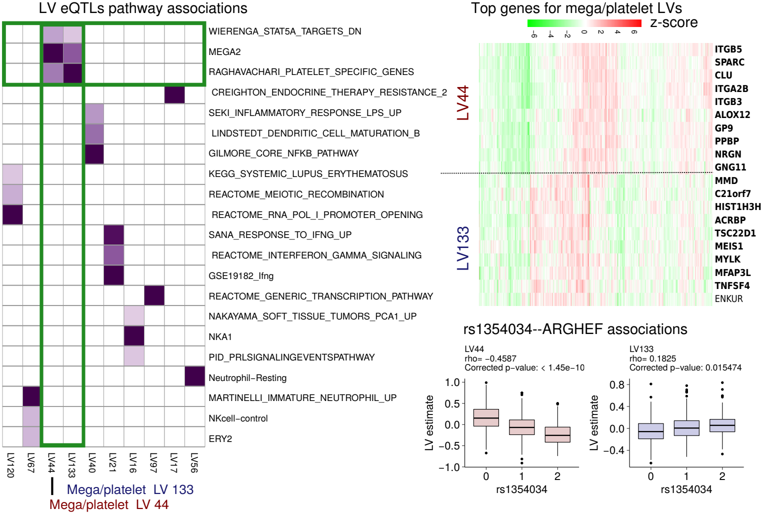


Top genes for mega/platelet LVs



LV133 (Mega/platelet LV **early**) genes are expression in megakaryocyte precursors.
LV44 (Mega/platelet LV **late**) genes are megakaryocyte specific.

A single locus controls 2 pathway effects



LV eQTLs pathway associations

Top genes for mega/platelet LVs

rs1354034--ARGHEF associations

LV133 (Mega/platelet LV **early**) genes are expression in megakaryocyte precursors.
LV44 (Mega/platelet LV **late**) genes are megakaryocyte specific.

Pleitropy of the ARGEF3 locus

- rs1354034 is known to be pleitropic: it affects both mean platelet volume (MPV) and platelet counts (PLT).

Furman-Niedziejko A. et al. Relationship between abdominal obesity, platelet blood count and mean platelet volume in patients with metabolic syndrome.

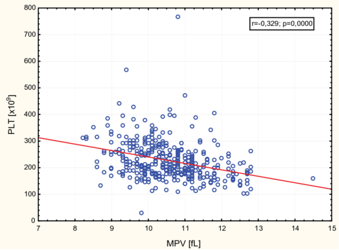
Pleitropy of the ARGEF3 locus

- rs1354034 is known to be pleitropic: it affects both mean platelet volume (MPV) and platelet counts (PLT).
- MPV and PLT are negatively correlated due to the tight control on total platelet volume.

Furman-Niedziejko A. et al. Relationship between abdominal obesity, platelet blood count and mean platelet volume in patients with metabolic syndrome.

Pleitropy of the ARGEF3 locus

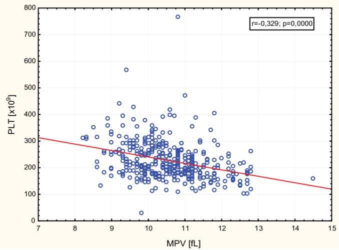
- rs1354034 is known to be pleitropic: it affects both mean platelet volume (MPV) and platelet counts (PLT).
- MPV and PLT are negatively correlated due to the tight control on total platelet volume.



Furman-Niedziejko A. et al. Relationship between abdominal obesity, platelet blood count and mean platelet volume in patients with metabolic syndrome.

Pleitropy of the ARGEF3 locus

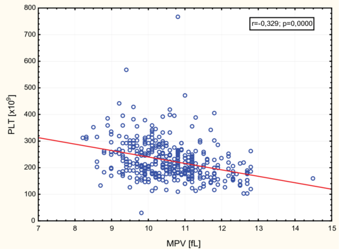
- rs1354034 is known to be pleitropic: it affects both mean platelet volume (MPV) and platelet counts (PLT).
- MPV and PLT are negatively correlated due to the tight control on total platelet volume.
- **Hypothesis:** LV133 (early) is associated with platelet number and LV44 (late) is associated with volume.



Furman-Niedziejko A. et al. Relationship between abdominal obesity, platelet blood count and mean platelet volume in patients with metabolic syndrome.

Pleitropy of the ARGEF3 locus

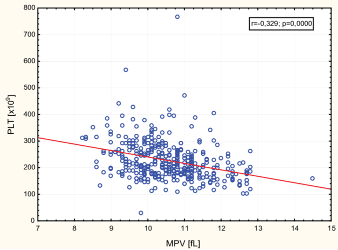
- rs1354034 is known to be pleitropic: it affects both mean platelet volume (MPV) and platelet counts (PLT).
- MPV and PLT are negatively correlated due to the tight control on total platelet volume.
- **Hypothesis:** LV133 (early) is associated with platelet number and LV44 (late) is associated with volume.
- We have data from large GWAS studies of blood count variables that show some loci regulate PLT and MPV independently. How are these associated with our latent variables?



Furman-Niedziejko A. et al. Relationship between abdominal obesity, platelet blood count and mean platelet volume in patients with metabolic syndrome.

Pleitropy of the ARGEF3 locus

- rs1354034 is known to be pleitropic: it affects both mean platelet volume (MPV) and platelet counts (PLT).
- MPV and PLT are negatively correlated due to the tight control on total platelet volume.
- **Hypothesis:** LV133 (early) is associated with platelet number and LV44 (late) is associated with volume.
- We have data from large GWAS studies of blood count variables that show some loci regulate PLT and MPV independently. How are these associated with our latent variables?



phenotype	reported SNP	Close gene	LV 133 p-value	LV44 p-value	proxy SNP
PLT	rs2911132	ERAP2	**2.4417e-05**	0.13817361	rs2549803
MPV	rs10876550	COPZ1	0.69933	**1.1847e-05**	rs10876550

Table: Raw p-values. 80 platelet related SNPs tested.

Furman-Niedziejko A. et al. Relationship between abdominal obesity, platelet blood count and mean platelet volume in patients with metabolic syndrome.

PLIER models transfer across datasets

Two human whole blood datasets:

- DGN: RNAseq US cohort
- NESDA: Affy European cohort

PLIER models transfer across datasets

Two human whole blood datasets:

- DGN: RNAseq US cohort
- NESDA: Affy European cohort

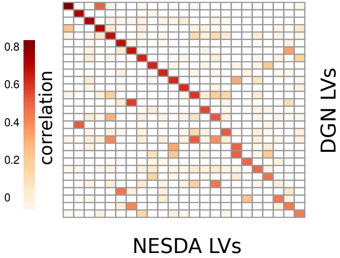
PLIER decompositions performed independently

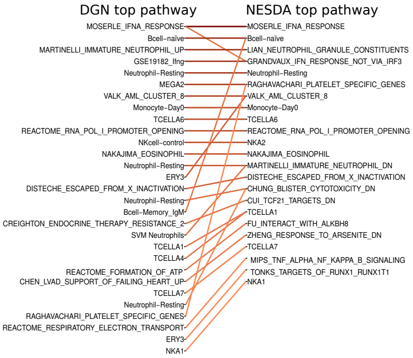
PLIER models transfer across datasets

Two human whole blood datasets:

- DGN: RNAseq US cohort
- NESDA: Affy European cohort

PLIER decompositions performed independently





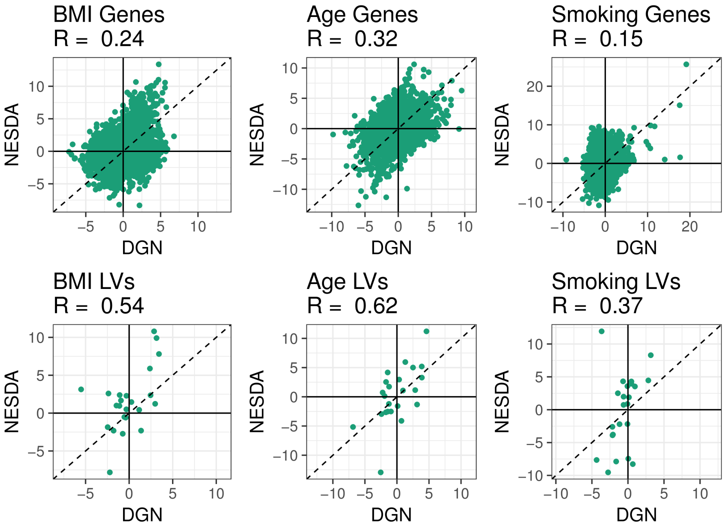
DGN top pathway

MOSERLE_IFNA_RESPONSE
Bcell-naïve
MARTINELLI_IMMATURE_NEUTROPHIL_UP
GSE19182_Ifng
Neutrophil-Resting
MEGA2
VALK_AML_CLUSTER_8
Monocyte-Day0
TCELLA6
REACTOME_RNA_POL_I_PROMOTER_OPENING
NKcell-control
NAKAJIMA_EOSINOPHIL
Neutrophil-Resting
ERY3
DISTECHE_ESCAPED_FROM_X_INACTIVATION
Neutrophil-Resting
Bcell-Memory_IgM
CREIGHTON_ENDOCRINE_THERAPY_RESISTANCE_2
SVM Neutrophils
TCELLA1
TCELLA4
REACTOME_FORMATION_OF_ATP
CHEN_LVAD_SUPPORT_OF_FAILING_HEART_UP
TCELLA7
Neutrophil-Resting
RAGHAVACHARI_PLATELET_SPECIFIC_GENES
REACTOME_RESPIRATORY_ELECTRON_TRANSPORT
ERY3
NKA1

NESDA top pathway

MOSERLE_IFNA_RESPONSE
Bcell-naïve
LIAN_NEUTROPHIL_GRANULE_CONSTITUENTS
GRANDVAUX_IFN_RESPONSE_NOT_VIA_IRF3
Neutrophil-Resting
RAGHAVACHARI_PLATELET_SPECIFIC_GENES
VALK_AML_CLUSTER_8
Monocyte-Day0
TCELLA6
REACTOME_RNA_POL_I_PROMOTER_OPENING
NKA2
NAKAJIMA_EOSINOPHIL
MARTINELLI_IMMATURE_NEUTROPHIL_DN
DISTECHE_ESCAPED_FROM_X_INACTIVATION
CHUNG_BLISTER_CYTOTOXICITY_DN
CUI_TCF21_TARGETS_DN
TCELLA1
FU_INTERACT_WITH_ALKBH8
ZHENG_RESPONSE_TO_ARSENITE_DN
TCELLA7
MIPS_TNF_ALPHA_NF_KAPPA_B_SIGNALING
TONKS_TARGETS_OF_RUNX1_RUNX1T1
NKA1

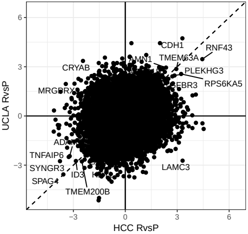
Correlation with phenotypes is more consistent in LV space



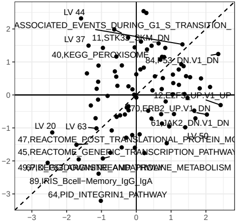
Some fun results

- Dataset from a collaborator: melanoma RNAseq , immunotherapy reponse (8 progressors, 11 responders).
- Very similart to the published Hugo et al. dataset [*] (13 progressors, 15 responders). How do they compare?

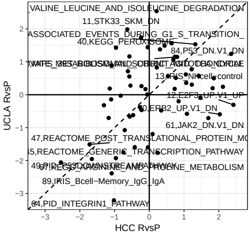
Spearman rho = 0.17



Spearman rho = 0.19



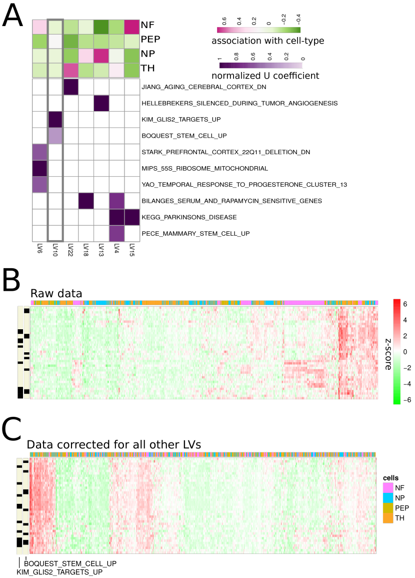
Spearman rho = 0.33



[*] Genomic and Transcriptomic Features of Response to Anti-PD-1 Therapy in Metastatic Melanoma

Usoskin et al. dataset

scRNAseq of mouse sensory neurons.



PLIER summary

- PLIER returns a set of latent variables that are both maximally independent from each other and maximally aligned with prior information.

PLIER summary

- PLIER returns a set of latent variables that are both maximally independent from each other and maximally aligned with prior information.

- Minimally supervised method: selects relevant pathways and discards thousands of irrelevant ones.

PLIER summary

- PLIER returns a set of latent variables that are both maximally independent from each other and maximally aligned with prior information.

- Minimally supervised method: selects relevant pathways and discards thousands of irrelevant ones.

- Additional output matrix U provides the mapping between pathways and LVs for quick interpretation.

PLIER summary

- PLIER returns a set of latent variables that are both maximally independent from each other and maximally aligned with prior information.

- Minimally supervised method: selects relevant pathways and discards thousands of irrelevant ones.

- Additional output matrix U provides the mapping between pathways and LVs for quick interpretation.

- Pathway-level estimates can be used in any subsequent analysis yielding mechanistic hypotheses.

Questions and future directions

- Group-level regularization on samples: not every LV exists in every sample.

Questions and future directions

- Group-level regularization on samples: not every LV exists in every sample.

- Looking for LVs that maximize objectives other than variance.

Questions and future directions

- Group-level regularization on samples: not every LV exists in every sample.
- Looking for LVs that maximize objectives other than variance.
- When are positivity constraints on the loadings necessary?

Acknowledgments

- University of Pittsburgh Computational and Systems Biology
  - **Wayne Mao**
- Icahn School of Medicine at Mount Sinai
  - Stuart Sealfon
  - Elena Zaslavsky
  - Boris Hartman
- Funding
  - 1R01 HG009299-01A1
  - U24 DK112331-01
  - U54 HG008540-03
  - 1R03 MH109009-01A1