USING A CROSS-TASK GRID OF LINEAR PROBES TO INTERPRET CNN MODEL PREDICTIONS ON RETINAL IMAGES

Anonymous authors
Paper under double-blind review

ABSTRACT

We analyze the relationships and shared structure among different prediction tasks on a dataset of retinal images using linear probes: linear regression models trained on some "target" task, using embeddings from a deep convolutional (CNN) model trained on some "source" task as input. We use this method across all possible pairings of 101 tasks in the UK Biobank dataset of retinal images, leading to ∼193k different models. We analyze the performance of these linear probes by source and target task and by layer depth.
We observe that representations from the middle layers of the network are more generalizable. We find that some target tasks are easily predicted irrespective of the source task, and that some other target tasks are more accurately predicted from correlated source tasks than from embeddings trained on the same task.
We then try to understand the principles that might be at work using synthetic experiments: images generated based on a "dead leaves" model.

1 INTRODUCTION

Retinal fundus (internal eye) images have been well-studied in machine learning applications. Machine learning can predict retinal disease with great accuracy (Gulshan et al., 2016; Badar et al., 2020). However, many other, often surprising, features can also be predicted from these images: for example, visual acuity (Varadarajan et al., 2018), cardiovascular risk (Poplin et al., 2018), diabetes (Zhang et al., 2021), anaemia (Mitani et al., 2020) and many other variables (Rim et al., 2020). Many of these are novel predictions not known to be predictable from these images by human experts, and it would be useful to understand precisely which features in the fundus image make these features predictable.

The challenge in even framing this question is the lack of tractable formalisms for characterizing *how* predictions are made. One simple idea is to take a model that achieves the surprising outcome of predicting certain variables from retinal images and ask what else this model is able to predict effectively. To do this, we need a way to evaluate how effectively we can build simple extensions of the models' internal representation to predict other quantities of interest. Linear probes (Alain & Bengio, 2016) and concept activation vectors (TCAV) (Kim et al., 2018) are techniques that provide paths to doing this; in this work we focus on linear probes. Linear probes have been widely used in a range of domains; we discuss their uses further in Section 4 on Related Work.

In this paper, we use this technique to study and understand CNN model predictions on 101 different tasks based on the UK Biobank dataset (Sudlow et al., 2015) of retinal images and labels.

We find that embeddings from the middle layers of the networks (as opposed to those closest to the output) learn features that are more generalizable across multiple target tasks, with linear probes consistently making accurate predictions. Additionally, we find that some target tasks such as eye position (left vs. right eye) and refractive error are easily predicted irrespective of the source task the CNN was trained on. We also find that other target tasks such as height are better predicted by embeddings trained on correlated tasks such as blood testosterone or self-reported sex, than by

embeddings trained on the original task. Ultimately these results give insight into which features of the input data make it possible to learn different target values.

2 RETINA IMAGES

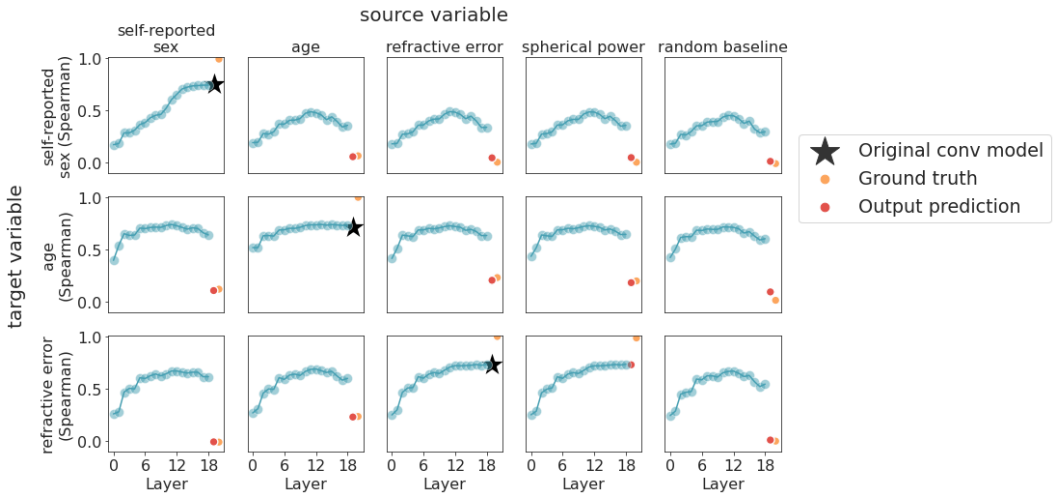


Figure 1: Plots of linear model performance by layer, organized by target variable (row) and source variable (column). Orange and red dots are from linear models trained on the single-valued source prediction and source variable value, respectively.

Data We used a dataset of retinal images from the UK Biobank study (Sudlow et al., 2015) containing 140,000 retinal fundus images from 68,000 patients. We randomly separated 12.5% of the patients into a test set, with the rest in the train set. We used a set of 101 non-eye-disease-related variables available in the UK Biobank resource. These included demographic data such as age and self-reported sex (recorded in the data as female/male); measurements such as blood tests and visual acuity; and miscellaneous features such as eye position (whether the image is of the left or right eye). Figure 4 summarizes all 101 variables.

2.1 EXPERIMENTS

We trained deep convolutional models with identical Inception V3 architectures for each of the 101 variables in our dataset. The models were pre-trained on ImageNet with auxiliary loss turned off and then trained with early stopping for a maximum of 200,000 steps. We then trained linear regression models for the same set of variables. Each model used the output of an intermediate layer from one of the convolutional models as input. We used 19 different intermediate layers spanning the depth of the Inception V3 architecture, and spatially average-pooled the output to make an exact linear regression tractable. These models were trained on the same training set as the convolutional models.

In total, this gave rise to $\sim$193k different linear models: 101 "source" tasks (convolutional models) $\times$ 101 "target" tasks (linear models) $\times$ 19 intermediate layers. We evaluated each linear model on the test dataset using Spearman's rank correlation.

Along with linear regressions on intermediate layers, we also carried out linear regressions on the raw values of the variables themselves and on the single-valued predictions from each convolutional model, in order to distinguish tasks that shared common representations from tasks that were merely correlated with each other. The regressions were done on the training set and evaluated on the test set, just like the intermediate layer regressions.

2.2 ANALYSIS AND OBSERVATIONS

In this section we present our analysis and observations from our experiments.

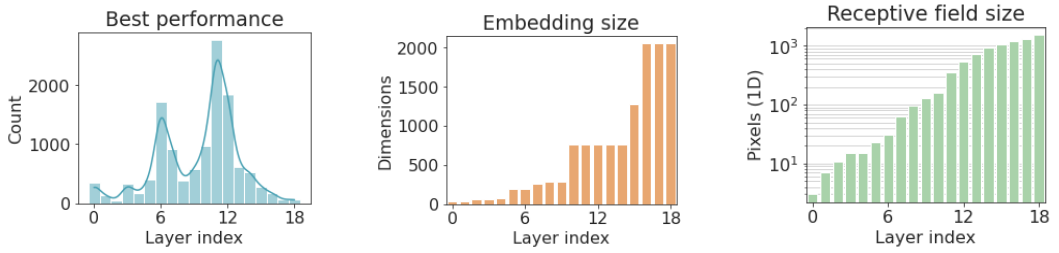
2.2.1 MIDDLE LAYER REPRESENTATIONS GENERALIZE BEST

Figure 1 presents performance of the linear probe models (y axis, as measured by Spearman's rank correlation) on embeddings across layers (x axis) from models trained to predict different variables (self-reported sex, eye position, smoker status, etc.) represented in each column. We observe that performance across layers tends to follow the same pattern: it increases from the layers closest to the input until the middle layers, and then decreases again - except where the source and target tasks are the same (on-diagonal plots). When the source and target are the same, the performance plateaus or continues to increase in the final layers.

The shape of the graph looks more similar along rows than columns, suggesting that the difficulty of learning a given task is more important than the differences between input embeddings learned for different tasks. This observation is in contrast with typical transfer learning setting where it is much more common to tune the final layers of a model (layers closer to the output) for new tasks, suggesting that we may want to entirely choose a layer in the middle for transfer learning.

In order to distinguish embeddings with similar representations from mere variable correlation, we also trained linear regression models on each input task's ground truth values and the convolutional model's output predictions (Fig 1, red and orange points). These generally do much worse than models trained on intermediate embeddings, though there are cases where the performance is comparable. One such example is predicting systolic blood pressure with age as the source task, which makes sense as the two variables are known to be correlated (Wolf-Maier et al., 2003).

2.2.2 SPECIFIC MIDDLE LAYERS PERFORM BEST, GENERALIZING WELL ACROSS MULTIPLE TASKS



(a) Histogram of the best-performing layer for each [source, target] variable pair. Pairs where the two variables were the same are excluded.

(b) Dimensions of embeddings (input to linear regression models) by layer of the source CNN.

(c) Receptive field (one dimension) by layer of the source CNN.

Figure 2

Next we ask, for the Inception network are there specific layers that give the best performance when generalizing to all tasks? Figure 2a shows a histogram of the best-performing layer for each [source, target] variable pair. (Pairs where the source and target tasks are the same were excluded.) It is interesting to see two distinct peaks in the middle layers (around layers 6,7 as well as layers 11, 12). However layer 11 appears to be consistently more generalizable and is amongst the top 3 layers with best performance on many of our pairwise comparisons. We don't know, however, why there are *two* peaks. The linear models' input does have different dimensions depending on the layer (Fig 2c), but the peaks don't obviously correspond to the changes in size. The clustering could also be due to correlations between [source, target] pairs. This would be interesting to investigate further.

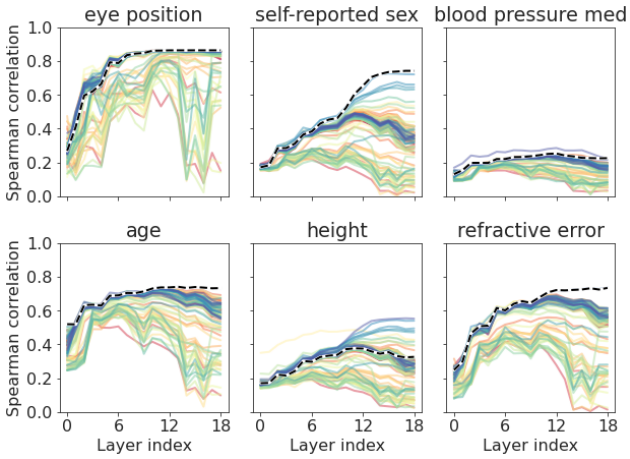


Figure 3: Comparison of performance on all source tasks for a given target task. Each line represents one source task. Color is based on the best performance of that source task as a *target* task - blue is highest performance, red is lowest. Dotted lines show the performance when the source variable is the same as the target. Observe that for 'height' representations from other source tasks (in this case, self-reported sex and blood testosterone) are better predictors.

2.2.3 Performance bands for same target task

As we would expect, the models generally perform best when the source and target tasks are the same (dotted lines, Fig 3). This is not always true, however - for example. "height" performs better on several other source tasks than on itself as a source task. The other sources were tasks such as as blood testosterone and self-reported sex, which share two properties: they're correlated with height, and are easier to predict than height (for which we never get a Spearman correlation above 0.4). We might guess that, because height is so hard to predict, there's not as much room for embeddings trained on it to improve - an illustration of the utility of multitask learning.

We can also observe that there are clear bands and outliers. The bands are closer together in the earlier layers, likely because features learned in the earlier layers are more universal. The outliers generally make sense: for example, the "blood pressure medication" as target task performs well with itself as a source task, but it performs just as well on source tasks for two related medications, aspirin and ACE inhibitors.

2.2.4 Comparison across all tasks

When comparing a single layer for all task combinations (Figure 4), we again see that the target task is usually more important than the source task. But there are some other interesting relationships - for example, age, eye position (left vs. right eye), and most measures of visual acuity are easily predicted regardless of the source task. Age is correlated with a small subset of tasks (Figure B12), but predicting age with those as source tasks gets only about the same performance as many other source task. These relationships can also be represented in a network specifying the pairs of tasks with high performance via linear probes, shown in Figure 5.

3 Understanding Linear Probes via Synthetic Images

Our experiments on the retina dataset highlight interesting correlations and hypothesis worth evaluating. To further study these correlations and hypotheses, we work on synthetic images where we have more control on the properties and tasks. We developed a set of environments of synthetic images where we focus on the shape, size, and number of objects, and create classification tasks in the spirit of the ones seen in the retinal dataset. This helps us answer our previous observations about why we see more than one peak when determining the best layer to transfer from, and why for a given target task, the model trained on the same source task is not always the best performing one.

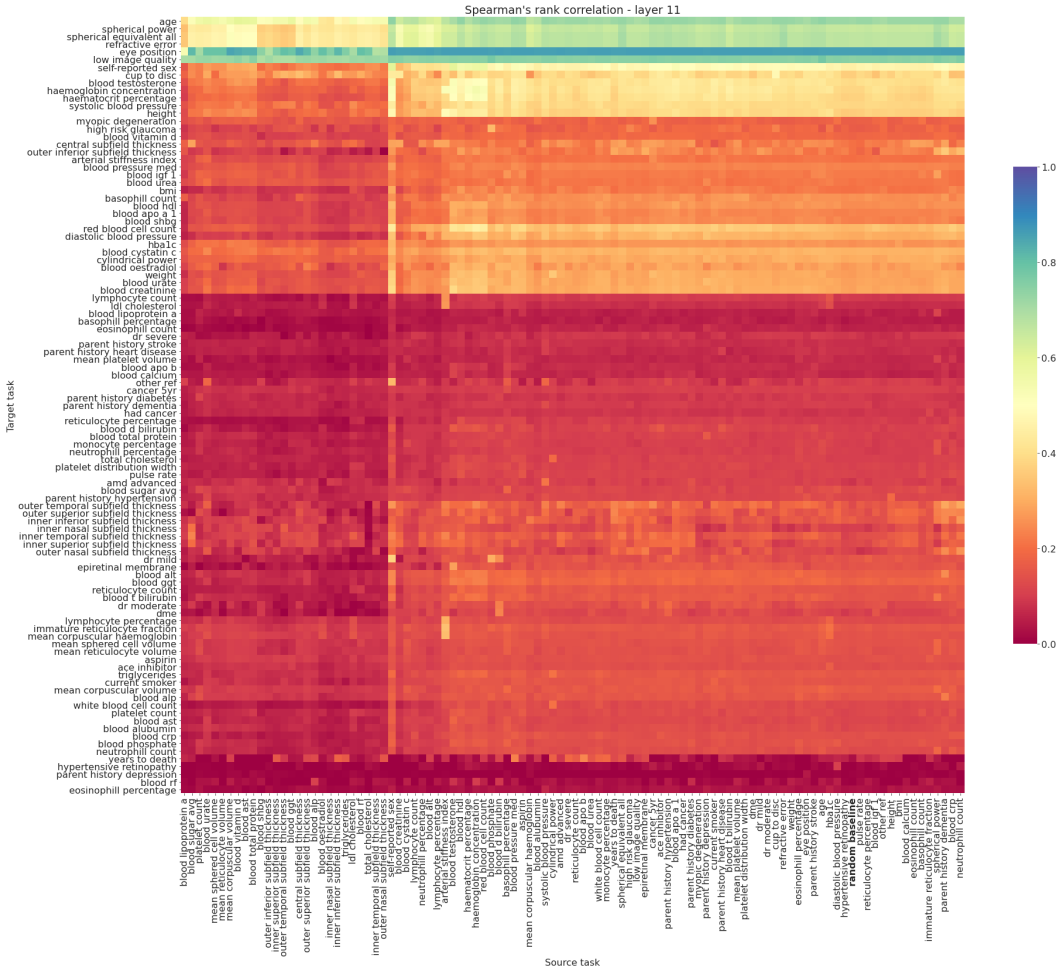


Figure 4: Cross-comparison of all tasks for layer 11. This layer was chosen as it was close to the best layer for most models, and showed the most interesting variation between variables. Tasks are ordered by hierarchical clustering. The bolded source task is a random baseline, where the source CNN was trained on labels drawn from the standard uniform distribution.

3.1 EXPERIMENTS

We generated images according to a "dead-leaves" model Matheron (1975); Lee et al. (2001) - so-called because it looks like fallen leaves piled on top of each other. Each dataset has 10000 images, all using "leaves" of a particular shape and size (Figure 6). The images are each the same size as the retina images, around 600x600 pixels.

Shape options:

- **Aligned squares**
- **Randomly rotated squares**

We also tried circles and 45°rotated squares, but the results didn't change much.

Size options:

- **Random (multi-size)**: For each side length in range(20, 600, 20), add one shape on the left side with probability 0.25, then repeat for the right side. Smaller shapes are always on top of larger ones. For tasks, instead of counting shapes of a given size, count shapes where n $\leq$ size $\leq$ n+50.

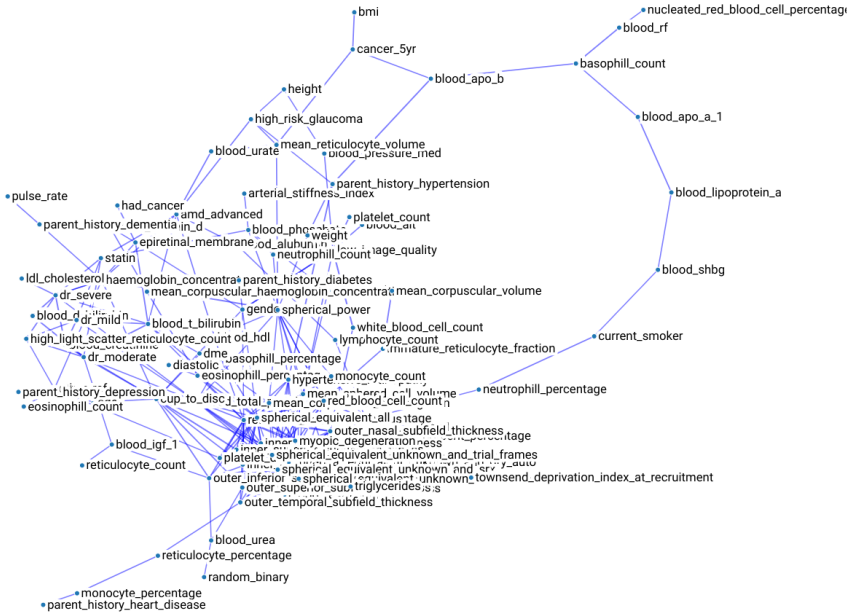


Figure 5: Network diagram of layer 11, connecting pairs of tasks with particularly good performance. The thickness of each line is determined by the ratio between the performance of a task pair on this layer (see Fig 4), and the performance of that pair when the "source" is the ground truth values. Lines are only shown if the ratio is above a threshold.

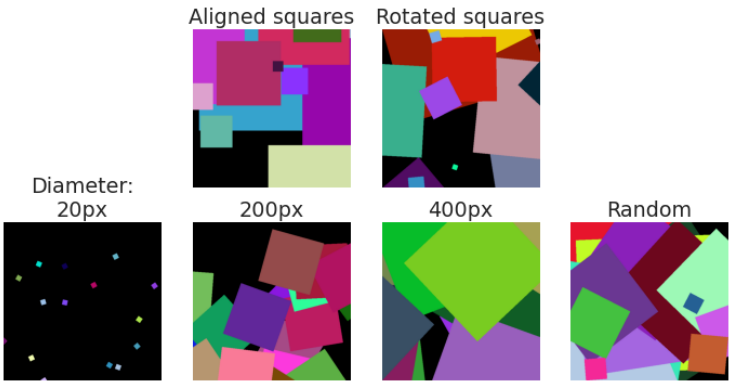


Figure 6: Shapes and size mixes used in synthetic image datasets.

- **20px / 200 / 400 (single-size)**: Same as for multi-size, except that instead of range(20, 600, 20), uses [n]*30 as the list of side lengths.

Within each dataset, there are 4 task types and up to 3 task sizes; each task involves counting all the shapes of a given size.

Task types:

- **Count all**: count total number of shapes of a given size
- **Count left/Count right**: count number of shapes of a given size, but only those in the left or right half of the image
- **Left > right?**: binary task - are there more shapes of the given size on the left or right half?

Task sizes: For each single-size dataset, we can of course only count shapes of that size. For multi-size datasets, we train each of the 4 task types for each of 3 "task sizes": 20px (counting all shapes in the 20-100 range), 200px (200-300), and 400px (400-500).

So in total, we have (2 shapes * 3 single-sizes * 4 task types * 1 task size) + (2 shapes * 1 multi-size * 4 task types * 3 task sizes) = 48 tasks.

The training procedure is the same as for the retina data: we train an Inception model for each of the 48 tasks, then train linear models on the intermediate layer outputs for the same set of tasks.

3.2 Major questions answered

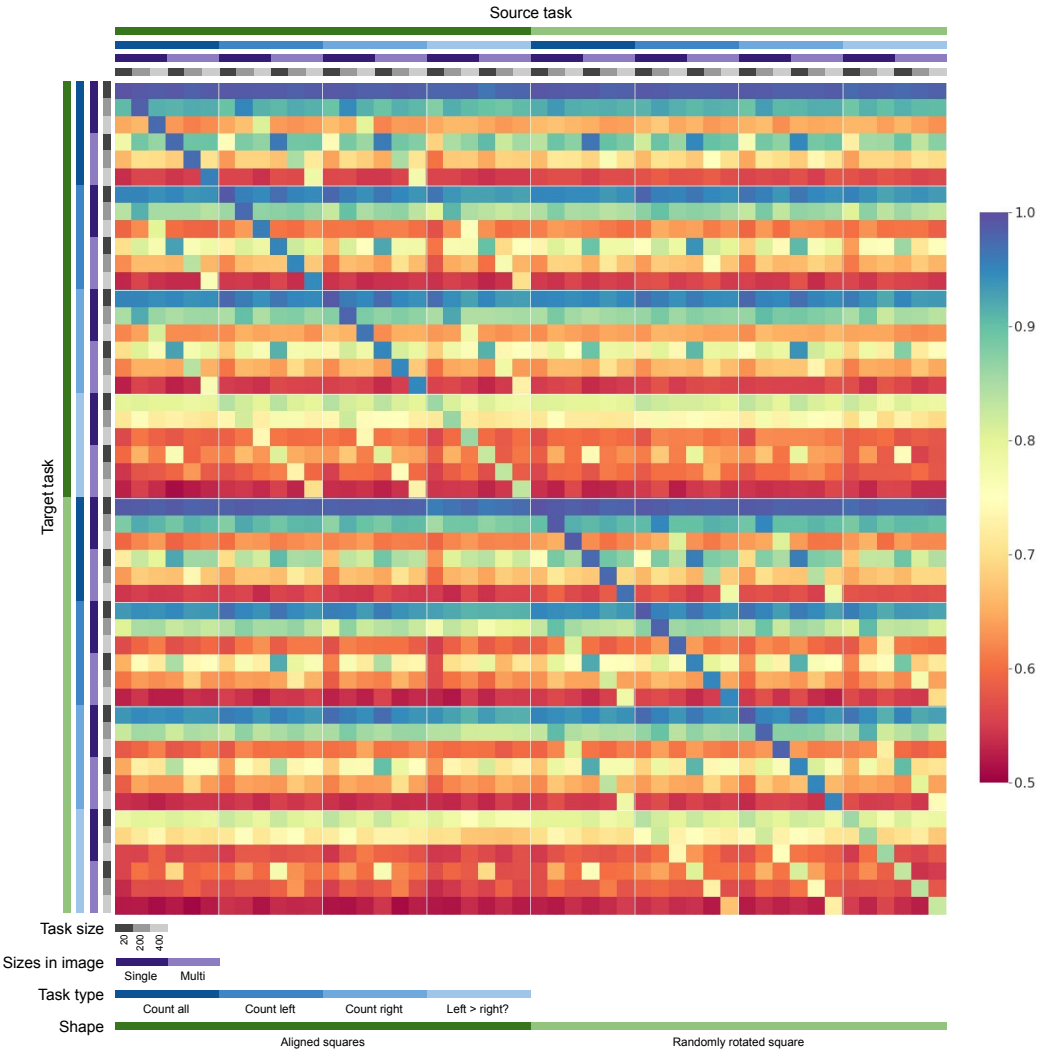
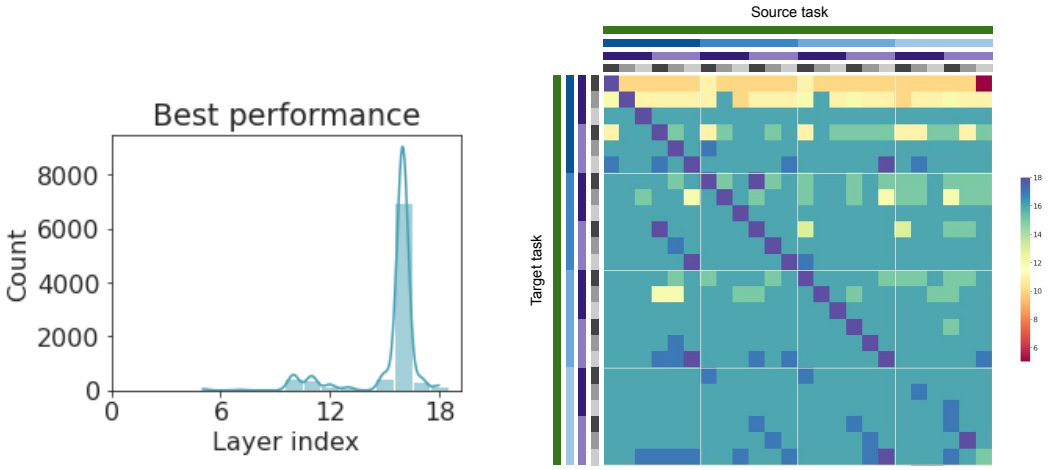


Figure 7: Cross-comparison heatmap of all tasks for layer 16.

3.2.1 Scale: Why are there two peaks in the best-layer histogram?

From Figure 8a, we can see that there are again two peaks in the layer histogram. Figure 8b shows that the first peak corresponds to target tasks with smaller features, likely because the necessary receptive field is smaller. Pairs where the source and target tasks are the same tend to do best at the highest layer, as expected.



(a) Histogram of the best-performing layer for each [source, target] task pair.

(b) Heatmap of the best layer for each task pair, for the subset of pairs where both tasks use aligned squares, i.e. upper left quadrant of Fig. 7. (The rest of the tasks look similar; see Fig. A11).

Figure 8: Best-performing layers for each task pair.

We might expect tasks that are correlated to peak at a higher layer than tasks that merely use shared representations. For example, "count left" is correlated with both "count all" and "left > right" for the same task size, but "count all" and "left > right" aren't correlated with each other, and neither is "count left" with "count right". We do observe a slight tendency for this to be true in Figure 7; there are some diagonals where the best layer is higher in the expected places.

3.2.2 BANDS: FOR A GIVEN TARGET TASK, WHY ISN'T THAT SAME TASK ALWAYS THE BEST SOURCE TASK?

We hypothesized that this phenomenon might correspond to source tasks that are more difficult for a CNN; if the CNN can't learn to make a good prediction, it will also be less likely to learn a good representation of the input. "Difficulty" might mean either there's not enough information in the input to make a good prediction, or the information is there but too complex to learn easily.

The former can be modeled by the information loss from shapes being occluded by others, especially the larger shapes, which in our data are always behind the smaller shapes. If we compare the tasks of counting 400-pixel-wide shapes in the single-size vs. multi-size examples, the single-size examples will likely be missing more information, since more shapes will be fully occluded by the large shapes piled on top. Meanwhile, we'd expect the multi-size tasks to be more complex, since we don't automatically know the size of each shape. Complexity is perhaps best modeled by the difference between shapes - we'd expect images with randomly rotated squares to be more difficult than similar images where all the squares have the same orientation.

We might expect that tasks with missing information would be worse source tasks than ones that are merely complex, since the model would be fitting to noise as well as real information; however, we don't see a significant effect in our results. More exploration would be useful, perhaps varying the numbers of shapes in each image to vary the amount of occlusion.

3.2.3 OTHER NOTABLE RELATIONSHIPS

Comparing target tasks, we can see that smaller task sizes are easier, which is unsurprising since occlusion is not really an issue. Smaller task sizes also have little dependence on source task. Single-size datasets are easier than multi-size ones.

The diagonal shows that for each target task, the best source task is itself. There are also other diagonal lines showing that the next best source tasks tend to be tasks with the same shape and task

size, and that learning is transferable between most task type pairs (with the exception of "count all" and "left > right").

Figure 9 shows that it's much easier to learn aligned squares from representations trained on rotated ones than the reverse.

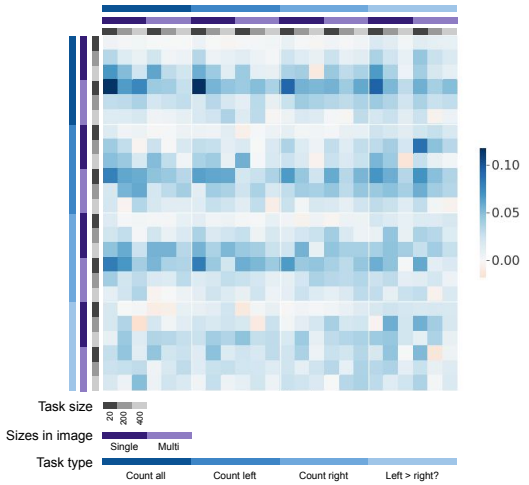


Figure 9: Difference between shapes as a source and target task on layer 16. Each point is the performance of a [source, target] pair where both source and target are aligned squares, minus the same pair where both source and target are rotated squares (i.e. quadrant I - quadrant III from Fig 7).

4 Related Work

Linear probes have been widely used for interpretability to understand performance of deep models with application to language processing (Hewitt & Liang, 2019; Hewitt & Manning, 2019; Belinkov, 2021), computer vision (Alain & Bengio, 2016; Asano et al., 2019), speech (Oord et al., 2018) or generally when understanding different neural network architectures (Raghu et al., 2017; Graziani et al., 2019; Horoi et al., 2020).

A number of recent papers on large vision, text and multimodal models Kolesnikov et al. (2019); Dosovitskiy et al. (2020); Radford et al. (2021) use linear probes as a way to verify the ability of the model's representations to transfer well to other datasets and tasks. Here we use linear probes to explain the model.

5 Conclusion

To conclude, in this work we used the basic notion of linear probes to study models trained on 101 different tasks/variables in fundus images in the UK Biobank. We looked at over 193k models, examining different variables as source and target pairs. We find interesting patterns in performance on source embeddings at various layer depths, and in performance on various combinations of source and target task.

Overall, the results show that simple linear probes provide a rich environment for unravelling the relationships between the underlying data and labels, providing insight into why neural networks trained on single labels are able to make accurate predictions. Future work will use the different representations to unravel which features of images are responsible for different accurate predictions.

6 Ethical Consideration

Understanding how representations are learned for medical applications can help design and develop better models for these applications, potentially helping to mitigate biases in model design. As

we apply these methods on medical domains and other sensitive domains we need to be mindful about the correlations and observations that we find. In particular we would want to evaluate the hypotheses that these interpretation methods yield and validate them with practitioners in the field.

REFERENCES

Guillaume Alain and Yoshua Bengio. Understanding intermediate layers using linear classifier probes. *arXiv preprint arXiv:1610.01644*, 2016.

YM Asano, C Rupprecht, and A Vedaldi. Self-labelling via simultaneous clustering and representation learning. In *ICLR*, 2019.

Maryam Badar, Muhammad Haris, and Anam Fatima. Application of deep learning for retinal image analysis: A review. *Computer Science Review*, 35:100203, 2020.

Yonatan Belinkov. Probing classifiers: Promises, shortcomings, and alternatives. *arXiv preprint arXiv:2102.12452*, 2021.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16$\times$ 16 words: Transformers for image recognition at scale. 2020.

Mara Graziani, Henning Muller, and Vincent Andrearczyk. Interpreting intentionally flawed models with linear probes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pp. 0–0, 2019.

Varun Gulshan, Lily Peng, Marc Coram, Martin C Stumpe, Derek Wu, Arunachalam Narayanaswamy, Subhashini Venugopalan, Kasumi Widner, Tom Madams, Jorge Cuadros, et al. Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs. *JAMA*, 316(22):2402–2410, 2016.

John Hewitt and Percy Liang. Designing and interpreting probes with control tasks. *arXiv preprint arXiv:1909.03368*, 2019.

John Hewitt and Christopher D Manning. A structural probe for finding syntax in word representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4129–4138, 2019.

Stefan Horoi, Guillaume Lajoie, and Guy Wolf. Internal representation dynamics and geometry in recurrent neural networks. *arXiv preprint arXiv:2001.03255*, 2020.

Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, et al. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In *International Conference on Machine Learning*, pp. 2668–2677. PMLR, 2018.

Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Joan Puigcerver, Jessica Yung, Sylvain Gelly, and Neil Houlsby. Big transfer (bit): General visual representation learning. *arXiv preprint arXiv:1912.11370*, 2019.

Ann B Lee, David Mumford, and Jinggang Huang. Occlusion models for natural images: A statistical study of a scale-invariant dead leaves model. *International Journal of Computer Vision*, 41 (1):35–59, 2001.

GFPM Matheron. Random sets and integral geometry. 1975.

Akinori Mitani, Abigail Huang, Subhashini Venugopalan, Greg S Corrado, Lily Peng, Dale R Webster, Naama Hammel, Yun Liu, and Avinash V Varadarajan. Detection of anaemia from retinal fundus images via deep learning. *Nature Biomedical Engineering*, 4(1):18–27, 2020.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

Ryan Poplin, Avinash V Varadarajan, Katy Blumer, Yun Liu, Michael V McConnell, Greg S Corrado, Lily Peng, and Dale R Webster. Prediction of cardiovascular risk factors from retinal fundus photographs via deep learning. *Nature Biomedical Engineering*, 2(3):158–164, 2018.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. *arXiv preprint arXiv:2103.00020*, 2021.

Maithra Raghu, Justin Gilmer, Jason Yosinski, and Jascha Sohl-Dickstein. SVCCA: Singular vector canonical correlation analysis for deep learning dynamics and interpretability. In *NIPS*, 2017.

Tyler Hyungtaek Rim, Geunyoung Lee, Youngnam Kim, Yih-Chung Tham, Chan Joo Lee, Su Jung Baik, Young Ah Kim, Marco Yu, Mihir Deshmukh, Byoung Kwon Lee, et al. Prediction of systemic biomarkers from retinal photographs: development and validation of deep-learning algorithms. *The Lancet Digital Health*, 2(10):e526–e536, 2020.

Cathie Sudlow, John Gallacher, Naomi Allen, Valerie Beral, Paul Burton, John Danesh, Paul Downey, Paul Elliott, Jane Green, Martin Landray, et al. UK Biobank: an open access resource for identifying the causes of a wide range of complex diseases of middle and old age. *PLOS Med*, 12(3):e1001779, 2015.

Avinash V Varadarajan, Ryan Poplin, Katy Blumer, Christof Angermueller, Joe Ledsam, Reena Chopra, Pearse A Keane, Greg S Corrado, Lily Peng, and Dale R Webster. Deep learning for predicting refractive error from retinal fundus images. *Investigative Ophthalmology & Visual Science*, 59(7):2861–2868, 2018.

Katharina Wolf-Maier, Richard S. Cooper, José R. Banegas, Simona Giampaoli, Hans-Werner Hense, Michel Joffres, Mika Kastarinen, Neil Poulter, Paola Primatesta, Fernando Rodríguez-Artalejo, Birgitta Stegmayr, Michael Thamm, Jaakko Tuomilehto, Diego Vanuzzo, and Fenicia Vescio. Hypertension Prevalence and Blood Pressure Levels in 6 European Countries, Canada, and the United States. *JAMA*, 289(18):2363–2369, 05 2003.

Kang Zhang, Xiaohong Liu, Jie Xu, Jin Yuan, Wenjia Cai, Ting Chen, Kai Wang, Yuanxu Gao, Sheng Nie, Xiaodong Xu, Xiaoqi Qin, Yuandong Su, Wenqin Xu, Andrea Olvera, Kanmin Xue, Zhihuan Li, Meixia Zhang, Xiaoxi Zeng, Charlotte L. Zhang, Oulan Li, Edward E. Zhang, Jie Zhu, Yiming Xu, Daniel Kermany, Kaixin Zhou, Ying Pan, Shaoyun Li, Iat Fan Lai, Ying Chi, Changuang Wang, Michelle Pei, Guangxi Zang, Qi Zhang, Johnson Lau, Dennis Lam, Xiaoguang Zou, Aizezi Wumaier, Jianquan Wang, Yin Shen, Fan Fan Hou, Ping Zhang, Tao Xu, Yong Zhou, and Guangyu Wang. Deep-learning models for the detection and incidence prediction of chronic kidney disease and type 2 diabetes from retinal fundus images. *Nature Biomedical Engineering*, 2021.

A HEATMAPS OF BEST LAYER FOR EACH TASK PAIR

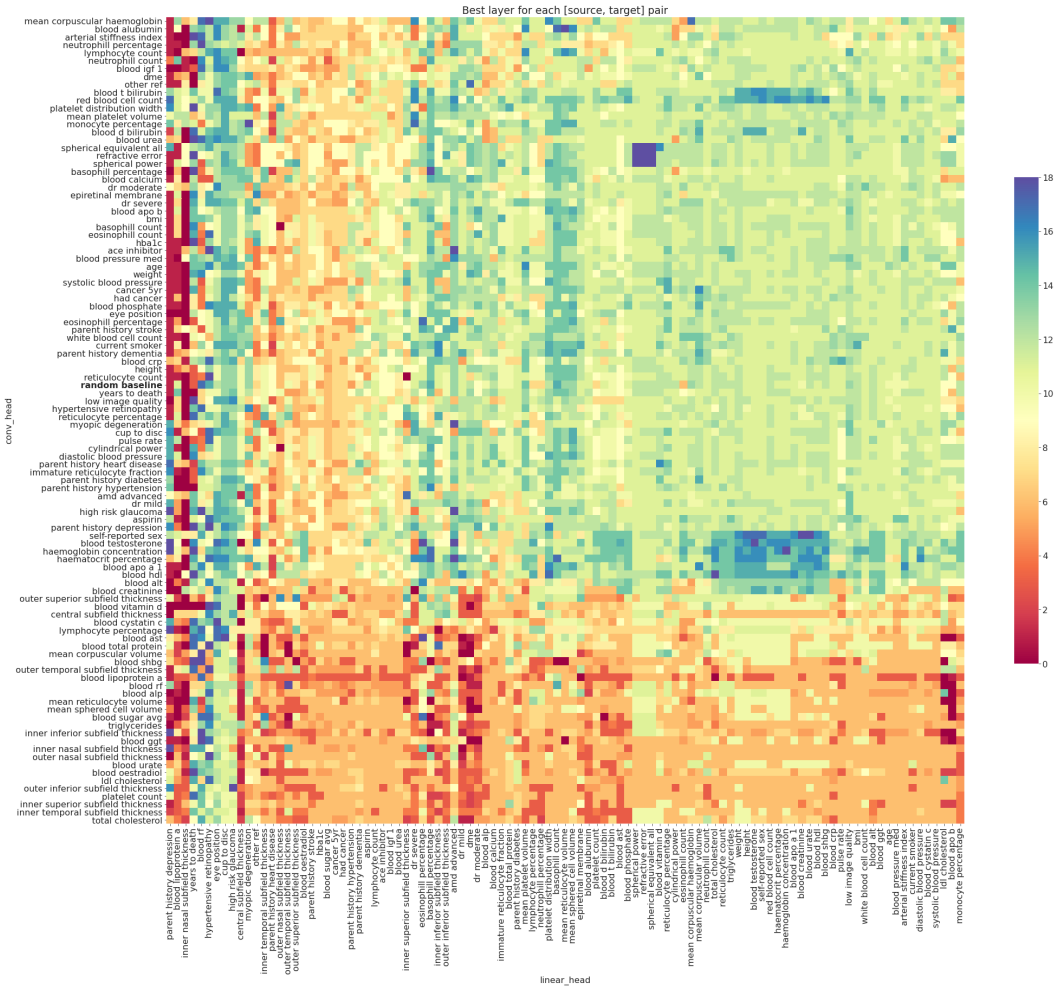


Figure 10: For each task pair, the layer of the source model that provided the best performance on the target task. 0 is earliest (closest to input), 18 is latest. Tasks are ordered by hierarchical clustering.

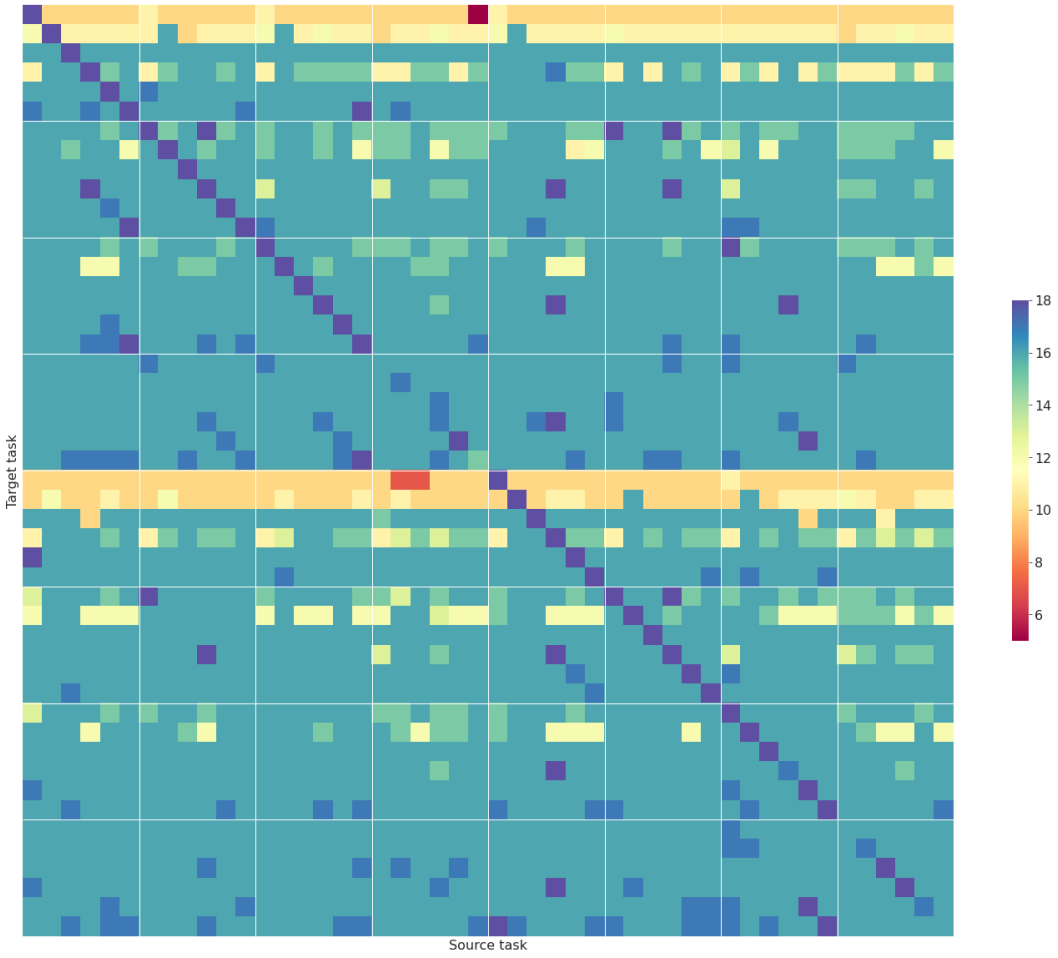


Figure 11: For each task pair, the layer of the source model that provided the best performance on the target task. 0 is earliest (closest to input), 18 is latest. See main paper for task legend.

B HEATMAPS FOR OTHER LAYERS, RETINA DATA

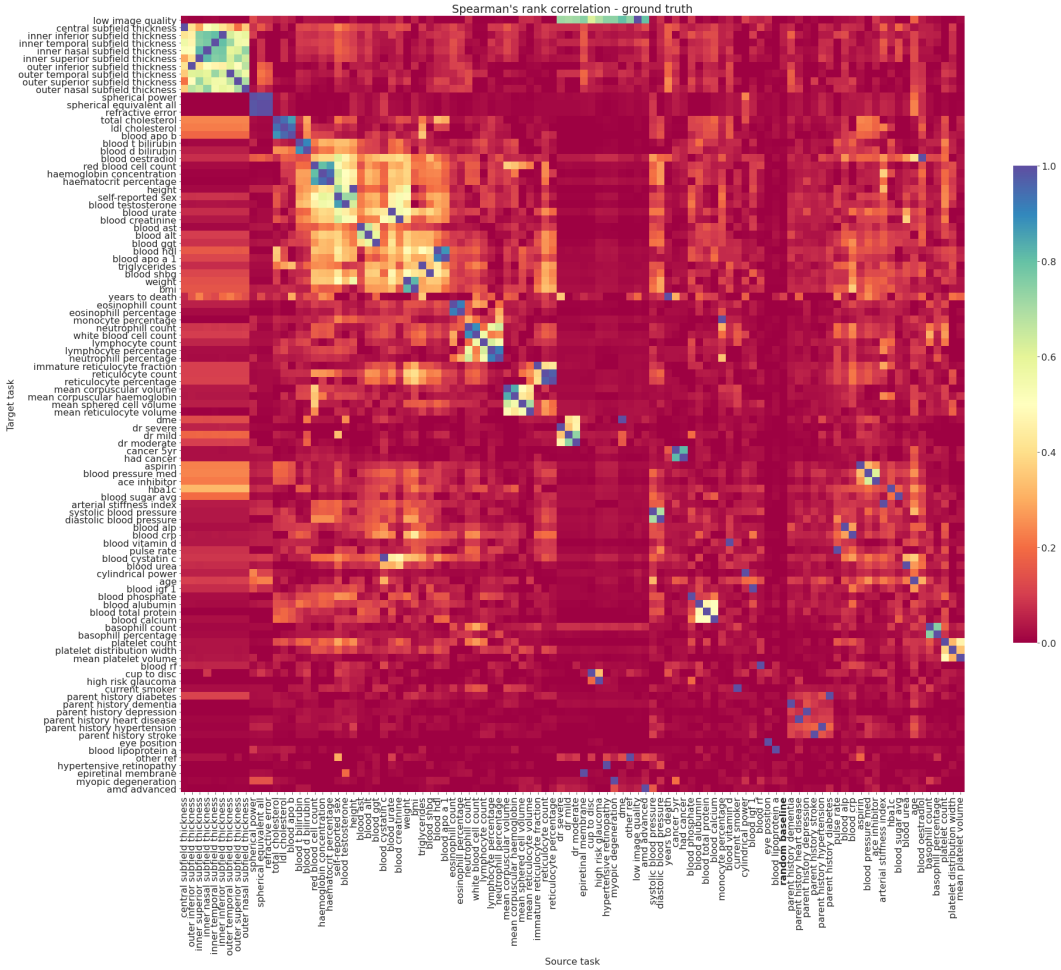


Figure 12: Cross-comparison of all tasks for linear models trained on the ground truth values for each "source task" (similar to simply calculating the correlation between the two tasks' ground truth values, except that we used a train/test split like we did for the other figures. Tasks are ordered by hierarchical clustering.)

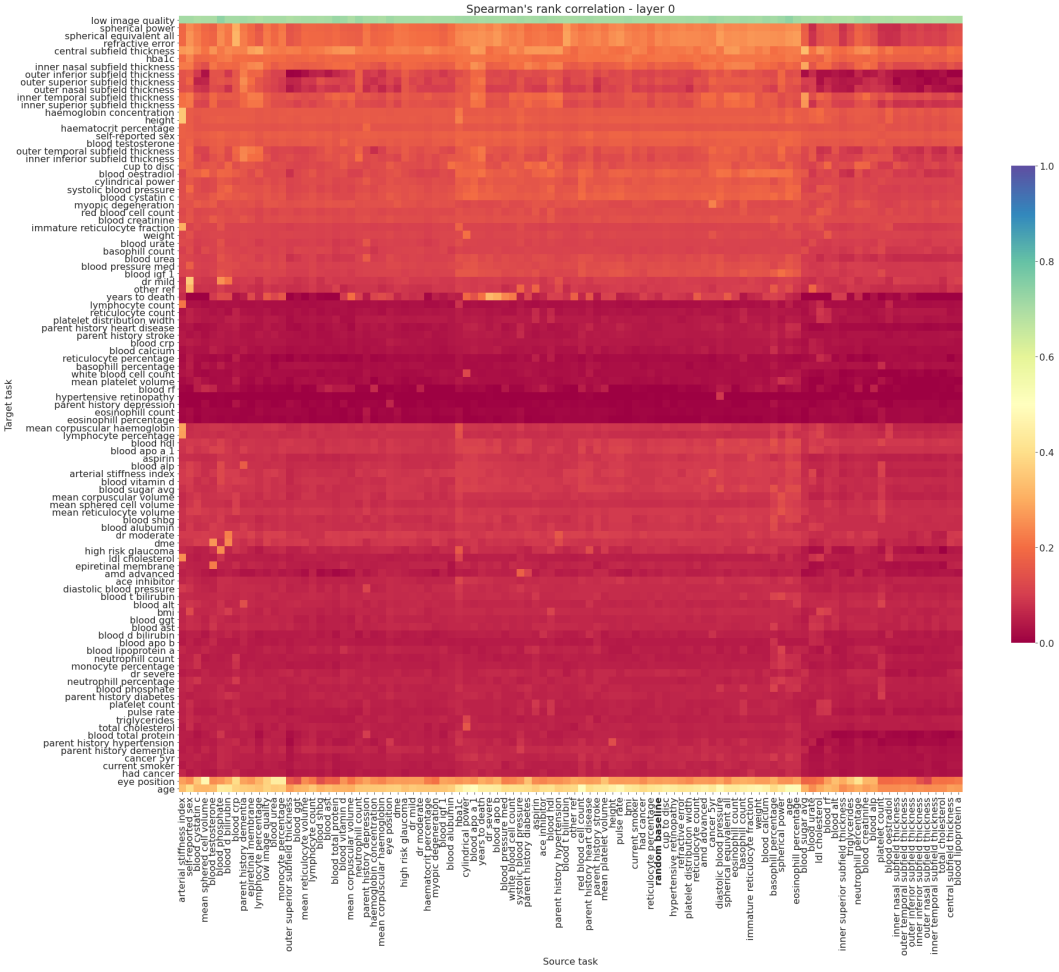


Figure 13: Cross-comparison of all tasks for different layers of the source model. Tasks are ordered by hierarchical clustering, which is done separately for each layer.

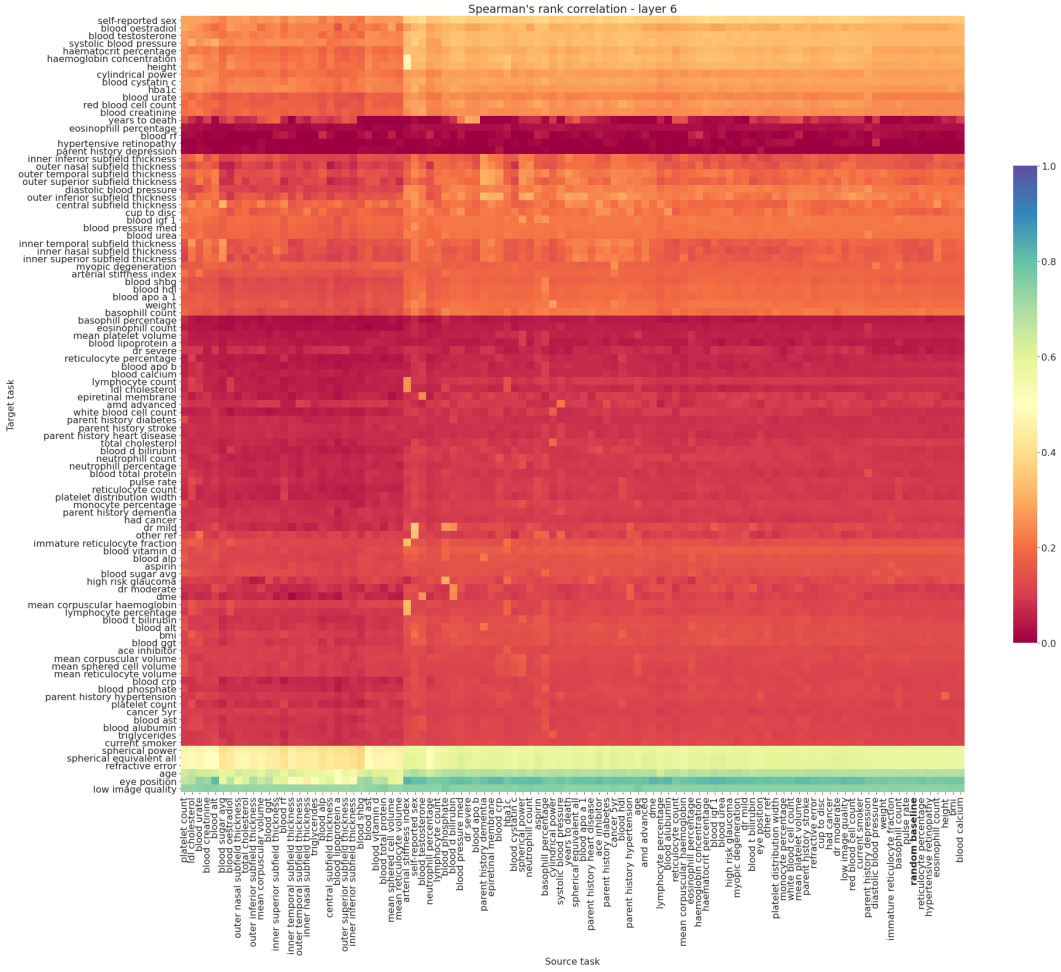


Figure 14: Cross-comparison of all tasks for different layers of the source model. Tasks are ordered by hierarchical clustering, which is done separately for each layer.

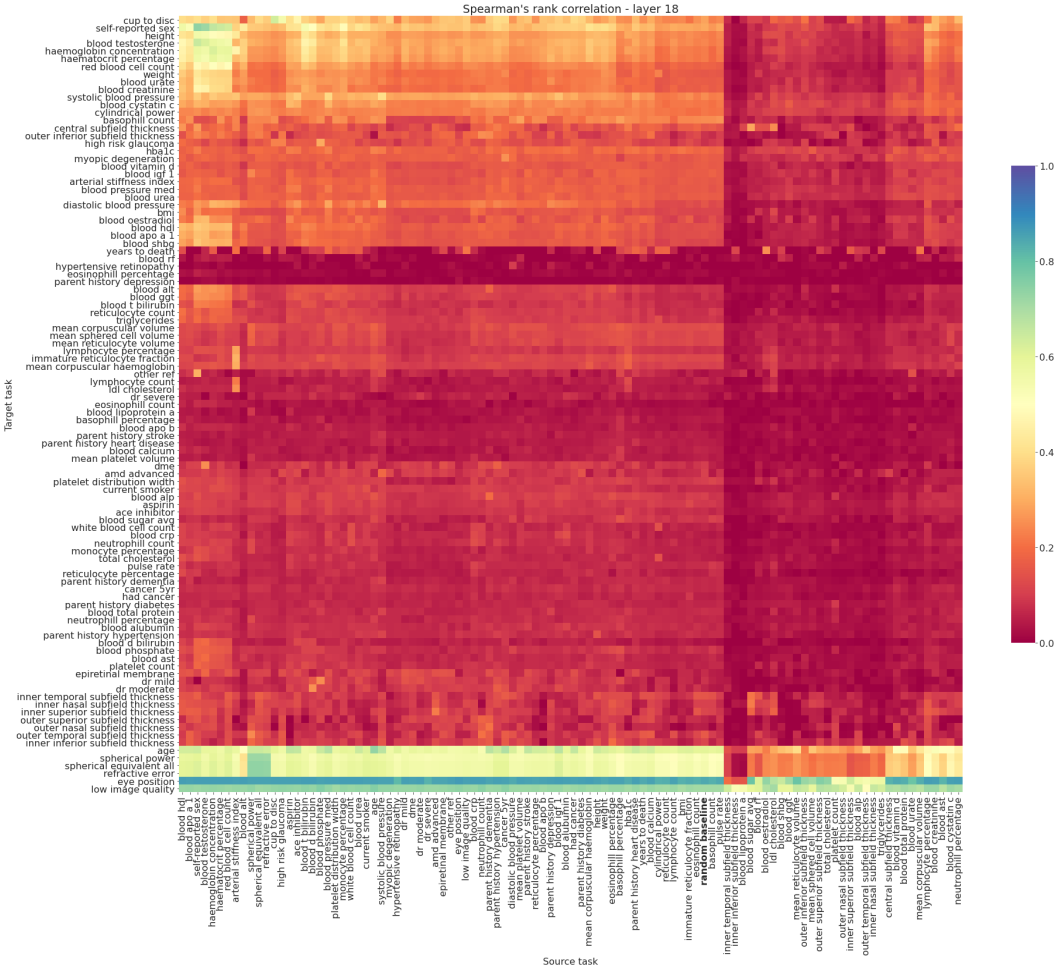


Figure 15: Cross-comparison of all tasks for different layers of the source model. Tasks are ordered by hierarchical clustering, which is done separately for each layer.