MULTI-DOMAIN LEARNING BY META-LEARNING: TAKING OPTIMAL STEPS IN MULTI-DOMAIN LOSS LANDSCAPES BY INNER-LOOP LEARNING

Anthony Sicilia\textsuperscript{1} Xingchen Zhao\textsuperscript{2} Davneet S. Minhas\textsuperscript{3} Erin E. O’Connor\textsuperscript{5}
Howard J. Aizenstein\textsuperscript{4} William E. Klunk\textsuperscript{4} Dana L. Tudorascu\textsuperscript{4} Seong Jae Hwang\textsuperscript{1,2}

\textsuperscript{1}Intelligent Systems Program - University of Pittsburgh
\textsuperscript{2}Computer Science - University of Pittsburgh
\textsuperscript{3}Radiology - University of Pittsburgh
\textsuperscript{4}Psychiatry - University of Pittsburgh
\textsuperscript{5}Department of Diagnostic Radiology & Nuclear Medicine - University of Maryland, Baltimore

ABSTRACT
We consider a model-agnostic solution to the problem of Multi-Domain Learning (MDL) for multi-modal applications. Many existing MDL techniques are model-dependent solutions which explicitly require nontrivial architectural changes to construct domain-specific modules. Thus, properly applying these MDL techniques for new problems with well-established models, e.g., U-Net for semantic segmentation, may demand various low-level implementation efforts. In this paper, given emerging multi-modal data (e.g., various structural neuroimaging modalities), we aim to enable MDL purely algorithmically so that widely used neural networks can trivially achieve MDL in a model-independent manner. To this end, we consider a weighted loss function and extend it to an effective procedure by employing techniques from the recently active area of learning-to-learn (meta-learning). Specifically, we take inner-loop gradient steps to dynamically estimate posterior distributions over the hyperparameters of our loss function. Thus, our method is model-agnostic, requiring no additional model parameters and no network architecture changes; instead, only a few efficient algorithmic modifications are needed to improve performance in MDL. We demonstrate our solution to a fitting problem in medical imaging, specifically, in the automatic segmentation of white matter hyperintensity (WMH). We look at two neuroimaging modalities (T1-MR and FLAIR) with complementary information fitting for our problem.

1. INTRODUCTION
In this paper, we consider the problem of Multi-Domain Learning (MDL) in which the goal is to take labeled data from some collection of domains \{\mathcal{D}_i\}_i and minimize the risk on \textit{all} of these domains. Note, this is in contrast to the related field of Domain Adaptation (DA) which minimizes risk on only a subset of these domains referred to as the target. Although our focus is MDL, it is not uncommon for Multi-Task Learning (MTL) solutions to be applicable to MDL problems. Where MDL assumes a collection of domains \{\mathcal{D}_i\}_i all paired with the same task \mathcal{T}, MTL assumes a collection of tasks \{\mathcal{T}_i\}_i paired with a single domain \mathcal{D} \cite{1}. One simple model-agnostic solution to both problems comes in the form of a weighted loss function used to learn new tasks without “forgetting” old tasks \cite{2}. This method can be simplified and adapted for the task of MDL by specifying loss functions for each domain and jointly training on all domains by optimizing for the convex combination (weighted average) of these loss functions. Inspired by this approach, our main contribution is to significantly build-upon this method by dynamically estimating the optimal weights of the convex combination throughout the training process. To achieve this, we appeal to the recently growing research area of learning-to-learn (or meta-learning) which uses the idea of hypothetical gradient steps taken during an inner-loop optimization to extract “meta-information” useful to the optimization task. Our method closely follows this idea to estimate a posterior distribution over the optimal weights of our loss function at each training iteration.

We showcase this method on a fitting problem in medical imaging, specifically, in the automatic segmentation of white matter hyperintensity (WMH) with multi-modal structural neuroimaging. Caused by various factors from neurological to vascular pathologies \cite{3}, WMH is prevalent in population of aging, e.g., Alzheimer’s disease (AD) \cite{4}. Typically, the automatic WMH segmentation task focuses on identifying hyperintense, or bright, white matter regions in T2-weighted fluid attenuated inversion recovery (FLAIR). However, FLAIR is often acquired for neurological disorders that directly search for strokes or lesions, whereas, in observational AD studies of our focus, FLAIR is much less common and T1-weighted Magnetic Resonance (T1-MR) image is the norm. Unfortunately, detecting WMH in T1-MR is extremely difficult since contained WMH regions severely lack contrast – a key feature for segmentation (see Fig. \ref{fig:example}) for the contrast difference and predictions). Hence, this setting provides a good opportunity for knowledge transfer across domains (T1-MR and FLAIR). While FLAIR may benefit from

Accepted to IEEE International Symposium on Biomedical Imaging 2021
the higher quantity of T1-MR samples in a given dataset, T1-MR may additionally benefit from the much higher quality of the FLAIR samples. Further, considering how common it is for patients to only have either T1-MR or FLAIR, MDL is particularly relevant in this case (rather than DA) to perform well on both domains (i.e., train with T1-MR and FLAIR, but predicts well given T1-MR only, Fig. 1E). In this paper, we present a solution for MDL within this context. Importantly, the approach is model-agnostic, making it easily applicable to a myriad of MDL problems besides WMH segmentation.

2. MULTI-DOMAIN LEARNING (MDL)

Several early works on MDL are ensemble-based, combining learned domain-specific parameters into a single classifier for inference [5 6]. More recent works separate shared parameters from domain-specific parameters via residual adapters [7] or a two-sided network [11]. Even adversarial approaches have been proposed [8] which requires separating the model parameters into a feature extractor and a task-specific network. Notably, at some level, these methods are all model-dependent, requiring explicit changes to the network architecture. This is less desirable if one wishes to enable MDL in segmentation since standard existing methods may not be trivially applicable to U-Net [9] (e.g., for adversarial approaches, one still needs to somehow define where feature-extraction stops and classification begins within the U-Net structure). Conversely, our approach is model-agnostic, making no model-dependent changes. It is therefore applicable to most existing models (including U-Net). This flexibility adds a great practical value for the end-user who wishes to enable MDL in a “plug-and-play” manner.

Learning to Learn. Learning-to-learn (meta-learning) is an algorithmic effort to not only learn some set of model parameters, but to learn the best way in which those model parameters can be learned. Many recent popularizations of this concept [10 11] – and formalizations [12] – largely involve an inner- and outer-loop. The dual-loop scheme uses the inner-loop to extract hypothetical model performance if the model were optimized in some way. From this, in the outer-loop, the hyperparameters of interest (e.g., the way the model is optimized) can be updated [10 11], or the model itself can be updated in a modified way [13]. While it is evident that meta-learning is an active area of research, our method focuses on the special case of a weighted loss function for MDL. Unlike many meta-learning solutions in the MTL problem space [11 14 15], we have only a single task, making it unclear how we could pre-train our hyperparameters as usual (i.e., using a distribution over tasks). To combat this, we exploit the functional form of our loss-function and use Bayesian estimation techniques to dynamically estimate our hyper-parameters during training (i.e., without any pre-training phase). One fallout of this, is an interesting differentiation of our approach from existing meta-learning literature. While the majority of meta-learning solutions are fully gradient based, our technique, instead, uses MAP estimation during inner-loop optimization.

3. PROPOSED APPROACH

We describe our approach (Alg. 1, Fig. 2) which can be applied universally to nearly any neural network model without model-specific changes. Our meta-learning procedure with outer- and inner-loop is as follows: (i) outer-loop updates the model parameters \( \theta \) based on (ii) inner-loop which learns and updates our hyperparameter (\( \lambda_0 \)). We first formalize the weighted loss function used in the outer-loop. Ultimately, we interpret this loss as an expectation over an optimal update choice, allowing us to learn \( \lambda_0 \) by MAP estimation.

3.1. Outer-loop Optimization of Model Parameter \( \theta \)

We define a domain \( \mathcal{D} = (\mathcal{X}, p(x)) \) as a feature space \( \mathcal{X} \) paired with a distribution of samples from that space \( p(x) \) [16]. For the remainder of the paper, we generally assume only two domains \( \mathcal{A} = (\mathcal{X}_A, p_A(x)) \) and \( \mathcal{B} = (\mathcal{X}_B, p_B(x)) \). We do this for brevity and for our two domain neuroimaging application, but in a subsequent section, we indeed show an easy extension to more than two do-
Further, we assume a single task \( T = (\mathcal{Y}, q(y)) \) (e.g., segmentation), a pre-specified model \( f \) (e.g., U-Net), and a possibly domain-specific loss function for both \( A \) and \( B \) written \( \mathcal{L}_A \) and \( \mathcal{L}_B \) respectively. The goal of our method is to dynamically determine the most appropriate weighting of these losses. Specifically, we seek \( \lambda_0 \) with \( 0 \leq \lambda_0 \leq 1 \) for the training objective below

\[
\lambda_0 \mathcal{L}_A(f(x^a; \theta), y^a) + (1 - \lambda_0) \mathcal{L}_B(f(x^b; \theta), y^b) \tag{1}
\]

where \( \theta \) is the current model parameters. The mini-batches \((x^a, y^a) \sim (p_A(x), q(y))\) and \((x^b, y^b) \sim (p_B(x), q(y))\) are (input,label) pairs from domains \( A \) and \( B \) respectively. In practice, this objective is achieved using a modified SGD to update \( \theta \). In particular, at step \( t \), we set \( \theta_{t+1} \) as below

\[
\theta_{t+1} \leftarrow \theta_t - \eta \nabla_{\theta_t} \left[ \lambda_t \mathcal{L}_A(f(x^a_t; \theta_t), y^a_t) + (1 - \lambda_t) \mathcal{L}_B(f(x^b_t; \theta_t), y^b_t) \right] \tag{2}
\]

with \( \eta \) the learning rate and mini-batches \((x^a_t, y^a_t)\) and \((x^b_t, y^b_t)\). Since Eq. (2) involves two losses, the learned \( \lambda_t \) weights the effect of gradients \( \nabla_{\theta_t} \mathcal{L}_A \) and \( \nabla_{\theta_t} \mathcal{L}_B \) which are best for \( A \) and \( B \) respectively. This **inner-loop** optimization (Alg. [1] line 10) differs from standard SGD with weighted losses since \( \lambda_t \) depends on the current \( \theta_t \) rather than being fixed. In the next section, we describe the inner-loop optimization step which estimates \( \lambda_t \) dynamically.

### 3.2. Inner-Loop Optimization of Hyperparameter \( \lambda_0 \)

**MAP Estimation of \( \lambda_0 \)**. We now discuss how to pick the optimal \( \lambda_0 \) at each time step. Here, the definition of optimal is fairly involved and formally analyzing how we define optimal requires a more complete picture. Subsequently, we assume for now that some notion of an optimal update choice is given and that this choice boils down to taking a step in the direction best for domain \( A \) (e.g., by using \( \nabla_{\theta_t} \mathcal{L}_A \)) or best for domain \( B \) (e.g., by using \( \nabla_{\theta_t} \mathcal{L}_B \)). Under this assumption, it is straightforward to interpret the multi-domain loss in Eq. (1) as an expectation over the optimal update choice (i.e., the expected best gradient to choose at time \( t \)). To see this, we assume during the update process there exists a sequence of (not necessarily i.i.d.) Bernoulli random variables indicating whether a step in the direction best for domain \( A \) or \( B \) is optimal. We can write the sequence \((\lambda_t)_t\) where \( t \) indexes over the sequential update process given in Eq. (2), \( \lambda_t \sim \text{Bernoulli}(\lambda_0) \), and \( \Lambda_t = 1 \) represents the event that taking a gradient step in the direction \( \nabla_{\theta_t} \mathcal{L}_A \) is optimal.

It then becomes simple to optimize \( \lambda_t \) dynamically by assuming a prior and updating sequentially with Maximum a Posteriori (MAP) Estimation. To meet the requirements of MAP Estimation, we make the simplifying assumption that the \( \lambda_t \) are i.i.d. in a small temporal window of size \( T \) (i.e., we perform our MAP updates using a history of length \( \leq T \)). Thus, we can, as usual\(^1\), assume the Beta\((\alpha, \beta)\) as our prior over \( \lambda_t \) and explicitly compute the MAP estimate (Alg.[1] line 8-9). Since the MAP estimate is precisely the mode of the posterior distribution, taking the log-likelihood and differentiating gives

\[
\lambda_t = \frac{\alpha + N_t - 1}{\alpha + \beta + T - 2} \tag{3}
\]

where \( T \) is the history length and \( N_t = \sum_{i=t-T}^{t} \Lambda_i \).

**Defining the Optimal Update Choice**. As alluded to, we still need define the optimal update choice. In particular, this implies we must define when \( \lambda_t = 1 \), or equivalently, when it is best to take a step in the direction \( \nabla_{\theta_t} \mathcal{L}_A \). With no prior knowledge, it is not clear when this should be the case. But, during training, we propose to effectively “explore” the local properties of our optimization space to gain the needed insight. This may be done using meta-learning. Specifically, during inner-loop phase, we can compare model performance after computing hypothetical gradient steps favoring \( A \) and \( B \), respectively. In the case of domain \( A \), we randomly split the mini-batch \((x^a_t, y^a_t)\) into a meta-train set \((\hat{x}^a_t, \hat{y}^a_t)\) and meta-test

---

\(^1\)usual, because a Beta Distribution is the Bernoulli Distribution’s conjugate prior

![Fig. 2. Proposed Approach.](image-url)
set \( \tilde{x}_t^a, \tilde{y}_t^a \) (Alg. 1 line 3). Then, we compute the hypothetical gradient favoring \( \Lambda \) (Alg. 1 line 4)
\[
\theta_t^a = \theta_t - \eta \nabla_{\theta_t} \mathcal{L}_A(f(\tilde{x}_t^a; \theta_t), \tilde{y}_t^a),
\]
and the hypothetical loss favoring \( \Lambda \) (Alg. 1 line 6)
\[
H_A^t = \mathcal{L}_A(f(\tilde{x}_t^a; \theta_t^a), \tilde{y}_t^a) + \mathcal{L}_B(f(\tilde{x}_t^a; \theta_t^a), \tilde{y}_t^a). \tag{5}
\]
We can similarly arrive at \( \theta_t^b \) and \( H_B^t \) (Alg. 1 line 5 and 7).

Given this information, we can more formally define two optimal update choices:

1. **greedy**: \( \Lambda_t = 1 \) if \( H_B^t > H_A^t \), otherwise \( \Lambda_t = 0 \)
2. **conservative**: \( \Lambda_t = 1 \) if \( H_A^t > H_B^t \), otherwise \( \Lambda_t = 0 \)

Note, the conservative definition is used in Alg. 1 line 8. To gain insight on these definitions, we must do some analysis. In particular, these hypothetical losses are functions of the model parameters, and so, we can analyze them by looking at the dominant terms in their Taylor Expansions (centered at \( \theta_t \)). Note, this type of analysis is common for interpretation of the inner-loop [17][13]. So, in applying the analysis to \( H_A^t \) evaluated at \( \theta_t^a \), the following is true\footnote{Since meta-train/test sets are simply samples drawn from the same distribution, we de-identify them in this expansion for interpretation.} for small enough \( \eta \)
\[
\begin{align*}
H_A^t &= \mathcal{L}_A(f(x_t^a; \theta_t), y_t^a) + \mathcal{L}_B(f(x_t^a; \theta_t), y_t^a) \\
&\quad - \eta \nabla_{\theta_t} \mathcal{L}_A(f(x_t^a; \theta_t), y_t^a) \nabla_{\theta_t} \mathcal{L}_A(f(x_t^a; \theta_t), y_t^a) \\
&\quad - \eta \nabla_{\theta_t} \mathcal{L}_B(f(x_t^a; \theta_t), y_t^a) \nabla_{\theta_t} \mathcal{L}_B(f(x_t^a; \theta_t), y_t^a) \\
&\quad + O(\eta^2).
\end{align*}
\]

Now, if we apply the same analysis to \( H_B^t \), we notice there are multiple common terms in the Taylor Expansions. So, if we ignore \( O(\eta^2) \) terms (which are small) and recognize the definition of the L2 norm, we have an approximation of \( H_A^t - H_B^t \) as below
\[
H_A^t - H_B^t \approx \eta ||\nabla_{\theta_t} \mathcal{L}_B(f(x_t^a; \theta_t), y_t^a)||_2^2 - \eta ||\nabla_{\theta_t} \mathcal{L}_A(f(x_t^a; \theta_t), y_t^a)||_2^2. \tag{7}
\]

Hence, in the **greedy** definition with \( H_B^t > H_A^t \), we can infer
\[
||\nabla_{\theta_t} \mathcal{L}_A(f(x_t^a; \theta_t), y_t^a)|| > ||\nabla_{\theta_t} \mathcal{L}_B(f(x_t^a; \theta_t), y_t^a)||. \tag{8}
\]
Likewise, in the **conservative** definition with \( H_A^t > H_B^t \), we can infer
\[
||\nabla_{\theta_t} \mathcal{L}_B(f(x_t^a; \theta_t), y_t^a)|| > ||\nabla_{\theta_t} \mathcal{L}_A(f(x_t^a; \theta_t), y_t^a)||. \tag{9}
\]

Since \( \lambda_t \) is the probability that \( \nabla_{\theta_t} \mathcal{L}_A \) is the optimal update choice, we see that the greedy definition prefers larger gradient steps, while the conservative definition prefers smaller.

**More than Two Domains.** Generalizing our approach to more than two domains is straightforward. Eq. (1) is extended to a convex combination with additional weights for each added domain. Next, the sequence of Bernoulli Distributions becomes a sequence of Multinomial Distributions whose conjugate prior is a Dirichlet; the MAP Estimate is still analytic. Lastly, the **optimal update choice** (Alg. 1 line 8) is defined by \( \text{argmax} \) instead of \( > \) and \( \text{argmin} \) instead of \( < \).

### 4. EXPERIMENTS

#### 4.1. Data and Preprocessing

We randomly selected \( N=20 \) older participants with WMH from our local normal aging AD study who were cognitively normal at the time of scan with mean age of 81.2 (s.d. = 7.15), 14 females, and a mean education of 14.2 (s.d. = 2.44) years. For each subject, we used a 3T Siemens Trio TIM scanner and 12-channel head coil to collect T1-MR (TE=2.98ms, TR=2.3s, FA=90°, \( 1 \times 1 \times 1.2 \text{mm voxel} \)) and FLAIR (TE=90ms, TR=91.6s, FA=150°, \( 1 \times 1 \times 3 \text{mm voxel} \)). For each pair of T1-MR and FLAIR, we used FSL [18] to process them in the following order: (a) spatially align T1-MR to FLAIR (212×256×48 dims), (b) N4-correction [19], (c) skull-strip using FSL BET, and (d) intensity normalize using WhiteStripe [20]. The ground-truth WMH in each FLAIR was labeled by a neuroradiologist on 5 continuous and identical slices across the subjects where WMH is common.

#### 4.2. Experiment Setup

We use two base networks: (i) the standard U-Net [9] and (ii) a light-weight (LW) variant of U-Net with 3% of the parameters and no pooling layers or skip-connects.

**Our Methods.** We setup our methods as described in Section 3 with FLAIR for \( A \) and T1-MR for \( B \). We try \( T = \{25, 100\} \) for both the greedy (Ours-G-T) and conservative (Ours-C-T) versions. We use a Beta(5,5) as our prior for \( \lambda_t \); this assumes equal likelihood for FLAIR/T1-MR to be optimal and imposes low likelihood of 0 or 1. Again, these are applied to the base models (U-Net, LW) without any architecture changes in a completely model-agnostic manner.

**Other Baselines.** The baselines are applied to both U-Net and LW as follows: (1) **F50-T50:** Fix the weighting of both FLAIR and T1-MR at 0.5 to treat them equally. This is the most naïve way to use any models without considering MDL. (2) **F10-T90:** Fix the weighting of FLAIR at 0.10 and T1-MR at 0.90, largely favoring T1-MR. (3) **F90-T10:** Fix the weighting of FLAIR at 0.90 and T1-MR at 0.10, largely favoring FLAIR. (4) **Simple:** Heuristically update the hyperparameter \( \lambda_t \) in Eq. (1) proportional to the difference of the hypothetical losses: \( \lambda_{t+1} = \lambda_t + \gamma (H_{FLAIR}^t - H_{T1}^t)/H_{FLAIR}^t \). We set Simple-G with \( \gamma = -0.1 \) and Simple-C with \( \gamma = 0.1 \) to heuristically mimic Ours-G and Ours-C respectively.

**Loss Function.** For both FLAIR and T1-MR we minimize the sum of the cross-entropy and dice score loss – a differ-
We proposed a model-agnostic solution to the problem of MDL. The solution is an extension of a simple weighted loss

Table 1. Means and standard deviations (s.d.) of metrics across all setups using two models (U-Net and LW) and two numbers of FLAIR subjects (12 and 8). -F and -T indicate the metrics are computed over FLAIR and T1-MR samples respectively. GAIN-µ is the total (summed) increase in DSC over the baseline F50-T50. GAIN-σ is the total decrease in s.d. of DSC from F50-T50. Results indicate our method increases DSC while reducing the s.d., in particular, when FLAIR data-availability is reduced.

| Method       | DSC-F | DSC-T | GAIN-µ | GAIN-σ |
|--------------|-------|-------|--------|--------|
| LW (12F)     |       |       |        |        |
| F50-T50      | 0.757 ± 0.011 | 0.360 ± 0.031 | 0.0    | 0.0   |
| F10-T90      | 0.729 ± 0.006 | 0.404 ± 0.026 | 0.016  | 0.010 |
| F90-T10      | 0.766 ± 0.008 | 0.278 ± 0.033 | -0.073 | 0.001 |
| Simple-G     | 0.740 ± 0.023 | 0.152 ± 0.072 | -0.225 | -0.053|
| Simple-C     | 0.714 ± 0.062 | 0.325 ± 0.050 | -0.078 | -0.070|
| Ours-G-25    | 0.758 ± 0.009 | 0.366 ± 0.029 | 0.007  | 0.004 |
| Ours-G-100   | 0.759 ± 0.010 | 0.375 ± 0.028 | 0.017  | 0.004 |
| Ours-C-25    | 0.758 ± 0.007 | 0.356 ± 0.025 | -0.003 | 0.010 |
| Ours-C-100   | 0.755 ± 0.008 | 0.351 ± 0.018 | -0.011 | 0.016 |

| U-Net (8F)   | DSC-F    | DSC-T | GAIN-µ | GAIN-σ |
|--------------|---------|-------|--------|--------|
| F50-T50      | 0.753 ± 0.008 | 0.361 ± 0.023 | 0.0    | 0.0   |
| F10-T90      | 0.725 ± 0.008 | 0.393 ± 0.026 | 0.004  | -0.003|
| F90-T10      | 0.766 ± 0.013 | 0.291 ± 0.030 | -0.057 | -0.012|
| Simple-G     | 0.738 ± 0.020 | 0.152 ± 0.055 | -0.224 | -0.044|
| Simple-C     | 0.716 ± 0.063 | 0.311 ± 0.081 | -0.087 | -0.113|
| Ours-G-25    | 0.755 ± 0.007 | 0.361 ± 0.023 | 0.002  | 0.001 |
| Ours-G-100   | 0.756 ± 0.010 | 0.368 ± 0.030 | 0.010  | -0.009|
| Ours-C-25    | 0.752 ± 0.007 | 0.355 ± 0.021 | -0.007 | 0.003 |
| Ours-C-100   | 0.753 ± 0.013 | 0.364 ± 0.035 | 0.003  | -0.017|

Fig. 3. Visualization of how the optimal update choice λt changes over time (U-Net 8F). Line shows mean. Band shows s.d. Early spikes favor more informative FLAIR samples.
which uses meta-learning with inner-loop MAP Estimation to dynamically learn the weights of our loss function. On a WMH segmentation problem, we show that our proposed method improves both performance and consistency in low resource scenarios. The approach is widely applicable for MDL, making no assumptions on the underlying model.

6. ACKNOWLEDGMENTS

This work was supported by the NIH/NIA (R01 AG063752, RF1 AG025516, P01 AG025204, K23 MH118070), and SCoUR Scholars Award. We report no conflicts of interests.

7. COMPLIANCE WITH ETHICAL STANDARDS

The study was performed in line with the principles of the Declaration of Helsinki. Approval was granted by the Ethics Committee of the University of Pittsburgh.

8. REFERENCES

[1] Yongxin Yang and Timothy M Hospedales, “A unified perspective on multi-domain and multi-task learning,” in ICLR, 2014.

[2] Zhizhong Li and Derek Hoiem, “Learning without forgetting,” IEEE transactions on pattern analysis and machine intelligence, vol. 40, no. 12, pp. 2935–2947, 2017.

[3] Patrick Vermersch, Jean Roche, Michèle Hamon, Christine Daems-Monpeurt, Jean-Pierre Pruvo, Philippe Demailly, and Henri Petit, “White matter magnetic resonance imaging hyperintensity in alzheimer’s disease: correlations with corpus callosum atrophy,” Journal of neurology, vol. 243, no. 3, pp. 231–234, 1996.

[4] Benjamin M Kandel, Brian B Avants, James C Gee, Corey T McMillan, Guray Erun, Jimit Doshi, Christos Davatzikos, David A Wolk, Alzheimer’s Disease Neuroimaging Initiative, et al., “White matter hyperintensities are more highly associated with preclinical alzheimer’s disease than imaging and cognitive markers of neurodegeneration,” Alzheimer's & Dementia: DADM, vol. 4, pp. 18–27, 2016.

[5] Mark Dredze and Koby Crammer, “Online methods for multi-domain learning and adaptation,” in Proceedings of the Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 2008, pp. 689–697.

[6] Mark Dredze, Alex Kulesza, and Koby Crammer, “Multi-domain learning by confidence-weighted parameter combination.” Machine Learning, vol. 79, 2010.

[7] Sylvester-Alvise Rebuffi, Hakan Bilen, and Andrea Vedaldi, “Learning multiple visual domains with residual adapters,” in Advances in Neural Information Processing Systems, 2017, pp. 506–516.

[8] Alice Schoenauer-Sebag, Louise Heinrich, Marc Schoenauer, Michele Sebag, Lani F Wu, and Steve J Altschuler, “Multi-domain adversarial learning,” in ICLR, 2019.

[9] Olaf Ronneberger, Philipp Fischer, and Thomas Brox, “U-net: Convolutional networks for biomedical image segmentation,” in MICCAI. Springer, 2015.

[10] Marcin Andrychowicz, Misha Denil, Sergio Gomez, Matthew W Hoffman, David Pfau, Tom Schaul, Brendan Shillingford, and Nando De Freitas, “Learning to learn by gradient descent by gradient descent,” in Neurips, 2016.

[11] Chelsea Finn, Pieter Abbeel, and Sergey Levine, “Model-agnostic meta-learning for fast adaptation of deep networks,” in ICML, 2017.

[12] Edward Grefenstette, Brandon Amos, Denis Yarats, Phu Mon Htut, Artem Molchanov, Franziska Meier, Douwe Kiela, Kyunghyun Cho, and Soumith Chintala, “Generalized inner loop meta-learning,” arXiv preprint arXiv:1910.01727, 2019.

[13] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales, “Learning to generalize: Meta-learning for domain generalization,” in AAAI, 2018.

[14] Sarah Bechtle, Artem Molchanov, Yevgen Chebotar, Edward Grefenstette, Ludovic Righetti, Gaurav Sukhatme, and Franziska Meier, “Meta-learning via learned loss,” arXiv preprint arXiv:1906.05374, 2019.

[15] Flood Sung, Li Zhang, Tao Xiang, Timothy Hospedales, and Yongxin Yang, “Learning to learn: Meta-critic networks for sample efficient learning,” arXiv preprint arXiv:1706.09529, 2017.

[16] Veronika Cheplygina, Marleen de Bruijne, and Josien PW Pluim, “Not-so-supervised: a survey of semi-supervised, multi-instance, and transfer learning in medical image analysis,” Medical image analysis, vol. 54, pp. 280–296, 2019.

[17] Alex Nichol, Joshua Achiam, and John Schulman, “On first-order meta-learning algorithms,” arXiv preprint arXiv:1803.02999, 2018.

[18] John Muschelli, Elizabeth Sweeney, Martin Lindquist, and Ciprian Crainiceanu, “fslr: Connecting the fsl software with r,” The R journal, vol. 7, no. 1, pp. 163, 2015.
[19] Nicholas J Tustison, Brian B Avants, Philip A Cook, Yuanjie Zheng, Alexander Egan, Paul A Yushkevich, and James C Gee, “N4itk: improved n3 bias correction,” IEEE transactions on medical imaging, vol. 29, no. 6, pp. 1310–1320, 2010.

[20] Russell T Shinohara, Elizabeth M Sweeney, Jeff Goldsmith, Navid Shiee, Farrah J Mateen, Peter A Calabresi, Samson Jarso, Dzung L Pham, Daniel S Reich, Ciprian M Crainiceanu, et al., “Statistical normalization techniques for magnetic resonance imaging,” NeuroImage: Clinical, vol. 6, pp. 9–19, 2014.

[21] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi, “V-net: Fully convolutional neural networks for volumetric medical image segmentation,” in 2016 Fourth International Conference on 3D Vision (3DV). IEEE, 2016, pp. 565–571.

[22] Fabian Isensee, Philipp Kickingereader, Wolfgang Wick, Martin Bendszus, and Klaus H Maier-Hein, “Brain tumor segmentation and radiomics survival prediction: Contribution to the brats 2017 challenge,” in International MICCAI Brainlesion Workshop. Springer, 2017, pp. 287–297.