Published in final edited form as:

Comput Vis ECCV. 2016 October ; 9907: 570–584. doi:10.1007/978-3-319-46487-9_35.

Abundant Inverse Regression using Sufficient Reduction and its Applications

Hyunwoo J. Kim[†], **Brandon M. Smith**[†], **Nagesh Adluru**, **Charles R. Dyer**, **Sterling C. Johnson**, and **Vikas Singh**

University of Wisconsin-Madison http://pages.cs.wisc.edu/~hwkim/projects/air

Abstract

Statistical models such as linear regression drive numerous applications in computer vision and machine learning. The landscape of practical deployments of these formulations is dominated by forward regression models that estimate the parameters of a function mapping a set of p covariates, $\boldsymbol{x}$, to a response variable, y. The less known alternative, Inverse Regression, offers various benefits that are much less explored in vision problems. The goal of this paper is to show how Inverse Regression in the "abundant" feature setting (i.e., many subsets of features are associated with the target label or response, as is the case for images), together with a statistical construction called Sufficient Reduction, yields highly flexible models that are a natural fit for model estimation tasks in vision. Specifically, we obtain formulations that provide relevance of individual covariates used in prediction, at the level of specific examples/samples — in a sense, explaining why a particular prediction was made. With no compromise in performance relative to other methods, an ability to interpret why a learning algorithm is behaving in a specific way for each prediction, adds significant value in numerous applications. We illustrate these properties and the benefits of Abundant Inverse Regression (AIR) on three distinct applications.

Keywords

Inverse regression; kernel regression; abundant regression; temperature prediction; Alzheimer's disease; age estimation

1 Introduction

Regression models are ubiquitous in computer vision applications (*e.g.*, medical imaging [1] and face alignment by shape regression [2]). In scientific data analysis, regression models are the default tool of choice for identifying the association between a set of input feature vectors (covariates) $\boldsymbol{x} \in \mathscr{X}$ and an output (dependent) variable $y \in \mathscr{Y}$. In most applications, the regressor is obtained by minimizing (or maximizing) the loss (or fidelity) function assuming the dependent variable y is corrupted with noise ε: $y = f(\boldsymbol{x}) + \varepsilon$. Consequently, solving for the regressor is, in fact, equivalent to estimating the expectation $\mathbb{E}[y|\boldsymbol{x}]$; in statistics and machine learning, this construction is typically referred to as *forward (or standard) regression* [3]. The above formulation does not attempt to model noise in $\boldsymbol{x}$

[†]Hyunwoo J. Kim and Brandon M. Smith are joint first authors.

directly. Even for linear forms of $f(\cdot)$, if the noise characteristics are not strictly additive and normally distributed (*i.e.*, so that $y = f(\boldsymbol{x} + \boldsymbol{\varepsilon}) \Leftrightarrow y = f(\boldsymbol{x}) + \boldsymbol{\varepsilon}'$), parameter estimates and consistency properties will not hold in general [4,5].

The above issues have long been identified in the statistics literature and a rich body of work has emerged. One form of these models (among several) is typically referred to as *inverse regression* [6]. Here, the main idea is to estimate $\mathbb{E}[\boldsymbol{x}|y]$ instead of $\mathbb{E}[y|\boldsymbol{x}]$, which offers asymptotic and practical benefits and is particularly suitable in high-dimensional settings. To see this, consider the simple setting in which we estimate a regressor for n samples of $\boldsymbol{x} \in \mathbf{R}^p$, $p \gg n$. The problem is ill-posed and regularization (*e.g.*, with ℓ_0 or ℓ_1 penalty) is required. Interestingly, for linear models in the inverse regression setting, the problem is still well specified since it is equivalent to a set of p univariate regression models going from y to a particular covariate x^j where $j \in \{1, \cdots, p\}$.

As an illustrative example, let us compare the forward and inverse regression models for $p > n$. For forward regression, we have $y = \mathbf{b}^\top \boldsymbol{x} + \boldsymbol{\varepsilon}$ and so $\mathbf{b}^* = (X^\top X)^{-1} X^\top \mathbf{y}$, where $X = [\boldsymbol{x}_1 \cdots \boldsymbol{x}_n]^\top$ and $\mathbf{y} = [y^1 \cdots y^n]^\top$. This is problematic due to a rank deficient $X^\top X$. But in the inverse regression case, we have $\boldsymbol{x} = \mathbf{b}y + \boldsymbol{\varepsilon}$ and so $\mathbf{b}^* = (\mathbf{y}^\top \mathbf{y})^{-1} \mathbf{y}^\top X$, which can be computed easily. In statistics, a widely used algorithm based on this observation is Sliced Inverse Regression (SIR) [3]. At a high level, SIR is a dimensionality reduction procedure that calculates $\mathbb{E}[\boldsymbol{x}|y]$ for each 'slice' (*i.e.*, bin) in the response variable y and finds subspaces where the projection of the set of covariates is dense. The main idea is that, instead of using the full covariance of the covariates $\boldsymbol{x}$'s or $[\boldsymbol{x}^\top\ y]^\top$, we use the $\mathbb{E}[\boldsymbol{x}|y]$ for each bin within y as a new feature with a weight proportional to the number of samples (or examples) within that specific bin. Then, a principal components-derived subspace for such a covariance matrix yields a lower-dimensional embedding that incorporates the proximity between the y's for subsets of $\boldsymbol{x}$.

This idea has seen renewed interest in the machine learning and computer vision communities [7,8]. For example, consider the following simple example demonstrated in a relatively recent paper [9]: Their goal was to utilize an intrinsic low-dimensional (*e.g.*, 2D or 3D) representation of the input image (or a silhouette) to predict body pose, which was parametrized as 3D joint angles at articulation points. Identifying the structure in the gram matrix of the output space enables identification of the conditional dependencies between the input covariates and the output multivariate responses. This is not otherwise possible. For example, we don't typically know which low-dimensional representation of the input images best predicts specific values of the output label.

The above discussion suggests that SIR models can effectively find a single global subspace for the input samples $\boldsymbol{x}$ considering the conditional distribution of $(\boldsymbol{x}|y)$. However, there are a number of practical considerations that the SIR model is ill-equipped to handle. For example, in computer vision applications operating in the wild, such as the temperature prediction task shown in Fig. 1, we can rarely find a global embedding that fully explains the relationship between the covariates and the response. In fact, subsets of samples may be differently associated with slices of the output space. Further, many 'relevant' features may be systematically corrupted or unavailable in a non-trivial fraction of images. In practice, one finds that these issues strongly propagate as errors in the calculated embedding, making the downstream analysis unsatisfactory. Of course, in the forward regression setting, this problem is tackled by performing feature selection via sparsity-type penalties, which emphasize the reliable features in the estimation. The direct application of this idea in the inverse regression model is awkward since the 'predictor' y (which is the response in forward regression) for $\boldsymbol{x}$ is just one dimensional.

It turns out that the desirable properties we seek to incorporate within inverse regression actually fall out of an inherent characteristic in many vision datasets, namely an *abundance* of features. In other words, in associating a large set of features derived from an image to an output label (or response) y, it is often the case that different subsets of features/covariates predict the label *equally well*. In the inverse regression context, this property enables adapting associations between density windows of the output space with *different subsets of covariates* dynamically on a sample-by-sample basis. If a covariate is generally relevant but missing for a small subset of examples (*e.g.*, due to occlusion, noise, or corruption), the formulation allows switching the hypothesis to a distinct 'support' of covariates for these samples alone.

In summary, exploiting abundance in inverse regression yields robust and highly flexible models. But perhaps more importantly, we obtain highly interpretable models (in an individual, sample-specific way), which is crucial in many applications. In a mammogram exam, for example, an explanation of why a patient was assigned a high probability of malignancy is critical for interpretability. With no compromise in performance, such functionality is valuable in applications but natively available in very few. Beyond Decision Trees and Inductive Logic Programming, regression models seldom yield such flexibility. Next, we give a few motivating examples and then list the main contributions of this paper.

1.1 Motivating Examples

Consider the two tasks in Fig. 1 where the relevance of features/covariates varies depending on the context and the specific samples under consideration. In facial age estimation, a feature from a local patch at a fiducial point (*e.g.*, lip, eye corners) carries a great deal of information for predicting age. But if the patch is occluded, this feature is not relevant for that particular image. Consider another example focused on ambient temperature estimation from outdoor scene images recently tackled in [10]. Here, we must deal not only with occlusion and corruption, but, depending on the context, the relevance of an otherwise predictive feature may also vary. For example, the appearance of a tree (*e.g.*, leaf color and density) may enable identifying subtle changes even within a specific season (*e.g.*, early or late spring). But in winter, after the trees have shed their leaves, this feature carries little useful information for predicting day-to-day temperature. In this case, the relevance of the feature varies with the specific values assigned to the response y. Importantly, being able to evaluate the different features driving a specific prediction can guide improvement of learning algorithms by enabling human interpretability.

1.2 Contributions

To summarize, *our contribution is a novel formulation using inverse regression and sufficient reduction that provides end-to-end statistical strategies for enabling (1) adaptive and* *dynamic associations between abundant input features and prediction outputs on an image-by-image basis, and (2) human interpretability of these associations*. Our model dynamically updates the relevance of each feature on a sample-by-sample basis and allows for missing or randomly corrupted covariates. Less formally, our algorithm explains *why* a specific decision was made for each example (based on feature-level dynamic weights). We analyze the statistical properties of our formulation and show experimental results in three different problem settings, which demonstrate its wide applicability.

2 Estimating the conditional confidence of covariates

Given a supervised learning task, our overall workflow consists of two main modules. We will first derive a formulation to obtain the confidence associated with individual covariates x^j conditioned on the label y. Once the details of this procedure are derived, we will develop algorithms that exploit these conditional confidences for prediction while also providing information on *which* covariates were responsible for that specific prediction. We start by describing the details of the first module.

2.1 A potential solution based on sufficient dimension reduction

The ideal mechanism to assign a confidence score to individual covariates, x^j, should condition the estimate based on knowledge of all other (uncorrupted) covariates x^{-j} as well as the response variable y. This is a combinatorial problem that quickly becomes computationally intractable. For example, even when we consider only a single pair of covariates and a response, the number of terms will quadratically increase as $f(x^1|x^2, y)$, $f(x^1|x^3, y)$, $f(x^1|x^4, y)$, … $f(x^1|x^p, y)$. Another related issue is that, when considering dependencies between multiple variables $f(x^1|x^2, x^3, \ldots, y)$, estimation is challenging because the conditional distribution is high-dimensional and the number of samples may be small in comparison. Further, in the prediction phase, we do not have access to the true y, which makes conditioning somewhat problematic. An interesting starting point in formulating a solution is the concept of *sufficient dimension reduction* [11], which is not widely known outside of statistics. We provide a definition and subsequently describe our idea.

Definition 1—*Given a regression model $h : X \to Y$, a reduction $\phi : \mathbf{R}^p \to \mathbf{R}^q$, $q \le p$, is sufficient for the regression task if it satisfies one of the following conditions:*

1. *inverse reduction,* $X|(Y, \phi(X)) \sim X|\phi(X)$,
2. *forward reduction,* $Y|X \sim Y|\phi(X)$,
3. *joint reduction,* $X \perp\!\!\!\perp Y|\phi(X)$,

where $\perp\!\!\!\perp$ *indicates independence*, ~ *means identically distributed, and $A|B$ refers to the random vector A given the vector B* [11,12].

Example 1—Suppose we are interested in predicting obesity y of a subject using a regression model $h : x \to y$ with 10 covariates such as weight x^1, height x^2, education x^3, age x^4, gender x^5, …, BMI x^{10}. Since obesity is highly correlated to weight and height, $(y|x^1, x^2, \ldots, x^{10}) \sim (y|\phi(x^1, \ldots, x^{10})) \sim (y|x^1, x^2)$. Here, we call $\phi : (x^1, \ldots, x^{10}) \to (x^1, x^2)$ a sufficient reduction for the given regression task. Also, for predicting BMI, *i.e.*, $h' : (x^1, \ldots, x^9, y) \to x^{10}$, $\phi' : (x^1, \cdots, x^9) \to (x^1, x^2)$ is a sufficient reduction since $(x^{10}|x^1, \ldots, x^9, y) \sim (x^{10}|x^1, x^2)$.

Our goal is to address the intractability problem by characterizing $(x^j|x^{-j}, y)$ in a simpler form based on the definition of sufficient reduction. Notice that sufficient reduction relies on specifying an appropriate regression model *and* we seek to derive identities for the expression $(x^j|x^{-j}, y)$. It therefore makes sense to structure our regression problem as $h : x^{-j}, y \to x^j$. The definition of forward reduction states that if $Y|X \sim Y|\phi(X)$ holds, $\phi(X)$ is a sufficient reduction for the regression problem h. In this definition, if we let $X = x^j$, $Y = (x^{-j}, y)$, and $\phi(X) = \phi(x^{-j}, y)$, we directly have $(x^j|x^{-j}, y) \sim (x^j|\phi(x^{-j}, y))$, as desired.

Why is this useful? The conditional distribution $f(x^j|x^1, \ldots, x^p) = f(x^j|\phi(x^{-j}))$ can be more efficiently estimated in a lower-dimensional space using sufficient reduction. In addition, once we make the assumption that the sufficient reduction function values coincide with y, *i.e.*, $\phi(x^{-j}) = y$, then estimating the conditional distribution simplifies to $f(x^j|x^1, \ldots, x^p) = f(x^j|\phi(x^{-j})) = f(x^j|y)$. Intuitively, this special case is closely related to the well-known conditional independence of features given a response used in a naïve Bayesian relationship:

$$f(y|x) \propto \frac{\prod f(x^j|y)f(y)}{f(x)}. \qquad (1)$$

In other words, given a sufficient reduction, all covariates x^j are conditionally independent. The form in Eq. (1) is simply a special case where $\phi(\cdot)$ is y; the general form, on the other hand, allows significant flexibility in specifying other forms for $\phi(\cdot)$ (*e.g.*, any lower-dimensional map) as well as setting up the conditional dependence concretely in the context of conditional confidence. Note that sufficient reduction methods are related to generative models (including Naïve Bayes). It is tempting to think that generative models with lower-dimensional hidden variables play the same role as sufficient reduction. However, the distinction is that the sufficient reduction ϕ from SIR can be obtained independently for any downstream analysis (regression) whereas hidden variables in generative models need to be specified and learned for each regression model. Now, the remaining piece is to give an expression for the conditional confidence distribution. For simplicity, in this work, we will use a multivariate Gaussian, which facilitates evaluating $\mathbb{E}[x^j|y]$ and $\text{VAR}[x^j|y]$ easily.

Remarks—Notice that x^j may not always correspond to a unique covariate. Instead, it may refer to a subset of covariates, *e.g.*, multiple features from a local patch in an image may constitute a specific x^j. In various practical situations it may turn out that one or more of these features may be irrelevant to the given regression problem. This situation requires special handling: briefly, we will consider the support of the regression coefficients for $\mathbb{E}[y|x^j]$ and measure the confidence of the feature by measuring the deviation from $\mathbb{E}[x|y]$ only along the related regression direction. These extensions will be described later.

2.2 A simple estimation scheme based on abundant features

The above description establishes the identity, $f(x^j|\phi(x^{-j})) = f(x^j|y)$, assuming $\phi(x^{-j}) = y$ and gives us a general expression to calculate the conditional confidence of individual covariates. What we have not addressed so far is a constructive scheme to actually calculate $\phi(x^{-j})$ so that it serves as a surrogate for y. We describe this procedure below based on sufficient reduction.

A natural strategy is to substitute y using predicted estimates, $\hat{y}$, derived from a subset of covariates, $\{1, \cdots, p\} \backslash j$. The difficulty, however, is that many of these subsets may be corrupted or unavailable. Fortunately, we find that in most situations (especially with image data), multiple exclusive subsets of the covariates can reliably predict the response. This corresponds to the *abundant* features assumption described earlier, and seems to be valid in many vision applications including the three examples studied in this paper. This means that we can define $\phi^I(x^I)$ for distinct subsets I of the covariate set, $\{1, \cdots, p\} \backslash j$. Intuitively, a potentially large number of I's will each index unique subsets and can eventually be used to obtain a reliable prediction for y, which makes the sufficient reduction condition, $\phi^I(x^I) = y$, sensible. Marginalizing over distinct I's, we can obtain $\mathbb{E}[x^j|\phi^I(x^I)]$ (described below). Then, by calculating the discrepancy between $\mathbb{E}[x^j|\phi^I(x^I)] = \mathbb{E}[x^j|\hat{y}^I]$ and x^j, we can evaluate the conditional confidence of each specific covariate x^j.

Marginalizing over I to calculate $\mathbb{E}[f(x^j|\phi^I(x^I))]$. To calculate $\mathbb{E}[f(x^j|\phi^I(x^I))]$, the only additional piece of information we need is the probability of the index set I. This can be accomplished by imposing a prior over each corresponding sufficient reduction, $\phi^I(\cdot)$, as $w_{\phi^I} \coloneqq \mathbb{E}[(y - \phi^I(x^I))^2]^{-1}$ which expresses the belief that the reliability of distinct sufficient reductions $\phi^I(\cdot)$ will vary as a function of the subset of patches it indexes.[2] This means that the conditional confidence for a covariate is calculated by a weighted mean of $f(x^j|\phi^i(x^i)) = f(x^j|\hat{y}^i)$ using w_{ϕ^j} (see Line 4 in Alg. 1). With these ingredients, we present the complete algorithm in Alg. 1.

Algorithm 1

Conditional Confidence of Feature Aware Regression

```
1: procedure Training
2:   Estimate a joint distribution for each covariate, $f(x^j, y)$
3:   Find sufficient reduction $\phi^I : x^I \rightarrow y$ for each subset of features $x^I$
4:   Estimate the prior/weight for $\phi_I(\cdot)$ as $w_{\phi^I} = \mathbb{E}[(y - \phi^I(x^I))^2]^{-1}$
5:   Estimate cond. confidence of feature $w_{x^j} \coloneqq \sum_I w_{\phi^I} f(x^j|\hat{y}^I) / \sum_I w_{\phi^I}$
6:   Fit a feature confidence aware regressor $h{:}[\{x^j\}_{j=1}^K, \{w_{x^j}\}_{j=1}^K] \rightarrow y$
7: procedure Prediction
8:   Evaluate $w_{x^j} \coloneqq \mathbb{E} f(x^j|\phi^I(x^I))$ by lines 3 and 5, with learned $w_{\phi^I}$.
```

[2] Recall that individual patches correspond to covariates, which will be univariate or multivariate depending on the descriptor we choose for the patch. Here, I indexes different subsets of patches.

```
9: $\hat{y}=h(\{x^j\}_{j=1}^{K},\{w_{x^j}\}_{j=1}^{K})$
```

2.3 Deriving priors for sufficient dimension reduction

We now describe how to derive priors for sufficient dimension reduction using a convex combination of multiple sufficient reductions. We assume that each weak sufficient reduction $\Phi^I(\cdot)$ is an unbiased estimator for y. Since a convex combination of unbiased estimators (expectation over estimators) is also an unbiased estimator, our problem is to find the optimal weights for such a combination of the sufficient reductions. Note that such an estimator will satisfy a minimum variance property. Once calculated, we will directly use the estimates as a prior for $\phi^I(\cdot)$.

Let $\phi^1(x^1)\sim\mathcal{N}(y,\sigma_1^2),\ldots,\phi^K(x^K)\sim\mathcal{N}(y,\sigma_K^2)$ denote a set of sufficient reductions for different subsets I in $\{1, \cdots, p\}\backslash j$ where I indices belong in the set $\{1, \cdots, K\}$. This means that $y = \mathbb{E}(\phi^I(x^I))$ since each estimator is unbiased. Note that each estimator is independent given y, which means, roughly speaking, the prediction errors among the different sufficient reductions are not correlated. So, the problem of calculating the weights, w, reduces to the following optimization model,

$$\min_{\boldsymbol{w}} \text{VAR}\left[\sum_{I=1}^{K}\phi^j(x^I)w^I\right] \text{ s.t.}\sum_{I}w^I=1 \text{ and } w^I\geq 0, \text{ for all } I\in 1,\ldots,K. \tag{2}$$

Since we assume that the error is independent given y, Eq. (2) can be written as

$$\min_{\boldsymbol{w}}\sum_{I=1}^{K}\sigma_I^2(w^I)^2 \text{s.t.}\sum_{I}w^I=1 \text{ and } w^I\geq 0, \text{ for all } I\in 1,\ldots,K \tag{3}$$

The optimal weights $\boldsymbol{w}$ have a closed form due to the following result.

Lemma 1—*Based on KKT optimality conditions, one can verify (see the extended paper) that the optimal weights for* Eq. (3) *are* $w^I=\sigma_I^{-2}/\sum_{k=1}^{K}\sigma_k^{-2}$. *This is a unique global optimum for* Eq. (3) *when* $\sigma_I^2>0$, $\forall I \in \{1, \ldots, K\}$.

This provides a weight for each subset $I \in \{1, \ldots, K\}$ for arbitrary constant K.

In the extended paper, we present2 a scheme to estimate the conditional confidence of specific features within a particular covariate by considering the sufficient reduction direction. This reduces the influence of irrelevant features within a multivariate covariate, given a regression task. Next, we introduce a variant of kernel regression when covariates (and their multivariate features) have an associated conditional confidence score.

3 Conditional confidence aware kernel regression

In this section we modify an existing kernel regressor formulation to exploit the conditional confidence of covariates. This final module is needed to leverage the conditional confidence towards constructions that can be applied to applications in machine learning and computer vision.

We start from the Nadaraya-Watson kernel regression with a Gaussian kernel. Since this estimator requires a dissimilarity measure between samples, we simply need to define a meaningful measure using the covariate confidences. To do so, we can use a simple adjustment such that the distance measure makes use of covariates (both univariate and multivariate) *differentially*, proportional to their confidence level. The expectation of distance of each pair of covariates weighted by confidence shown below is one such measure:

$$\mathrm{d}_w(x_1, x_2, w_1, w_2) := \sqrt{\frac{\sum_j w_{x_1^j} w_{x_2^j} \left(x_1^j - x_2^j\right)^2}{\sum_j w_{x_1^j} w_{x_2^j}}}. \quad (4)$$

The expression in Eq. (4) can be interpreted as agnostic of the example-specific labels (even if they were available). Interestingly, the weights w_{x^j} are obtained via a surrogate to the unknown labels/responses via sufficient reduction. This scheme will still provide meaningful distances even when one or more covariates are corrupted or unavailable. Next, we modify Eq. (4) so we can guarantee that it is an unbiased estimator for distances between uncorrupted covariates under some conditions.

3.1 Unbiased estimator for distance between uncorrupted covariates

This section covers a very important consequence of utilizing inverse regression. Notice that it is quite uncommon in the forward regression setting to derive proofs of unbiasedness for distance estimates in the presence of corrupted or missing covariates or features. This is primarily because few, if any, methods directly model the covariates x^j. Interestingly, inverse regression explicitly characterizes $f(x^j|\phi^I(x^I))$, which means that we have access to $\mathbb{E}[x^j|x^{-j}]$. Let us assume that the 'true' but unobserved value of the covariate is $z^j \approx \mathbb{E}[x^j|x^{-j}]$. Since our model assumes that x^j is observed with noise, we can model the variance of x^j given $\mathbb{E}[x^j|x^{-j}]$ using $\sigma^2_{x^j|z^j} = \mathbb{E}[(x^j - \mathbb{E}[x^j|x^{-j}])^2]$, i.e., $x^j \sim \mathcal{N}(z^j, \sigma^2_{x^j|z^j})$. This allows us to obtain a powerful "corrected" distance measure. We now have:

Proposition 1—*Assume that we observe covariates x^1, x^2 with Gaussian noise given ground truth feature values z^1 and z^2, i.e., $x_1^j \sim \mathcal{N}(\overline{z}_1^j, \sigma^2_{x^j|z^j})$ and $x_2^j \sim \mathcal{N}(\overline{x}_2^j, \sigma^2_{x^j|z^j})$. Then, we have*

$$\mathbb{E}[(x_1-x_2)^2]=\mathbb{E}[x_1]^2+\mathbb{E}[x_2]^2-2\mathbb{E}[x_1]\mathbb{E}[x_2]-2\mathrm{COV}(x_1,x_2)+\mathrm{VAR}[x_1]+\mathrm{VAR}[x_2]$$
$$=\overline{x}_1^2+\overline{x}_2^2-2\overline{x}_1\overline{x}_2+2\sigma_{x|z}^2=(\overline{x}_1-\overline{x}_2)^2+2\sigma_{x|z}^2 \quad (5)$$

Thus, $(x_1-x_2)^2-2\sigma_{x|z}^2$ *is an unbiased estimator for distances between true (but unobserved) covariate values, e.g.,* $(z_1-z_2)^2=\mathbb{E}[(x_1-x_2)^2-2\sigma_{x|z}^2]$.

Once we have access to $2\sigma_{x|z}^2$, deriving the unbiased estimate simply involves a correction. So, we obtain the corrected distances:

$$\mathrm{d}(x_1,x_2,w_1,w_2)^2 := \mathbb{E}_j\left[\left((x_1^j-x_2^j)^2-2\sigma_j^2\right)\right]=\frac{\sum_j((x_1^j-x_2^j)^2-2\sigma_j^2)w_{x_1^j}w_{x_2^j}}{\sum_j w_{x_1^j}w_{x_2^j}}. \quad (6)$$

4 Results and discussion

Our method is broadly applicable, and so we show results on three different computer vision datasets, each with an associated task: 1) outdoor photo archives for temperature prediction, 2) face images for age estimation, and 3) magnetic resonance imaging (MRI) of brains for Alzheimer's disease prediction. For temperature prediction on the *Hot or Not* dataset [10], we show that are algorithm can help explain *why* a specific prediction was made without sacrificing accuracy compared to the state-of-the-art. We use age estimation as a familiar example to demonstrate several properties of our approach, namely that our global (w_{ϕ}^{j}), and dynamic weights (w_{x}^{j}) are meaningful and intuitive. Finally, we show that our method can be used to pinpoint regions of the brain image that contribute most to Alzheimer's disease prediction, which is valuable to clinicians.

4.1 Temperature prediction

Hot or Not [10] consists of geo-located image sequences from outdoor webcams (see supplement). The task is to predict ambient outdoor temperature using only an image of the scene. For fair comparison, we evaluated our method on the same 10 sequences selected by [10]. Like [10], we used the first-year images for training and the second-year images for testing.

We decompose temperature T into a low-frequency component T_{lo} and a high-frequency component T_{hi} as in [10]. We train our algorithm to predict T_{lo} and T_{hi} separately, and then estimate the final temperature as $T = T_{\mathrm{lo}} + T_{\mathrm{hi}}$. Intuitively, T_{lo} is correlated with seasonal variations (*e.g.*, the position of the sun in the sky at 11:00am, the presence or absence of tree leaves) and T_{hi} is correlated with day-to-day variations (*e.g.*, atmospheric conditions).

Glasner *et al.* [10] demonstrated good performance using each pixel and color channel as a separate feature. Our approach assumes a set of consistent landmarks across the image set. In principle, we could treat each pixel and color channel as a 'landmark,' but doing so would result in impractically slow training. Therefore, we adopt a two-level (hierarchical) approach.

We first describe our lowest-level features. Let $z_t = I_{i,j,c,t}$ be the image intensity at pixel *i, j*, color channel $c \in$ {red, green, blue, gray}, and time $t \in \mathcal{T}$. Let T_t be the ground truth temperature at time *t*. We omit the lo/hi subscript below. Each pixel produces a temperature estimate according to a simple linear model, $\hat{T}_{i,j,c,t} = a_{i,j,c} z_t + b_{i,j,c}$, where $\hat{T}_{i,j,c,t}$ is the estimated temperature at time *t* according to pixel *i, j, c, t*. We compute the regression coefficients $a^* = a^*_{i,j,c}$ and $b^* = b^*_{i,j,c}$ by solving $a^*, b^* = \min_{a,b} \sum_{t \in \mathcal{T}} \|az_t + b - T_t\|^2_2$. A straightforward way to produce a single prediction is to combine the pixel-wise predictions using a weighted average, $\hat{T}_t$. We form two feature vectors at each pixel, $\mathbf{t}_{i,j,t} = [\hat{T}_{\text{red}}, \hat{T}_{\text{green}}, \hat{T}_{\text{blue}}, \hat{T}_{\text{gray}}]$ corresponding to temperature estimates, and $\mathbf{v}_{i,j,t} = [z_{\text{gray}}, g_x, g_y]$, where z_{gray} is the grayscale pixel intensity and g_x and g_y are the *x* and *y* grayscale intensity gradients, respectively.

We divide the image into non-overlapping $h \times w$-pixel patches and assign a landmark to each patch (we empirically set $h = w = 15$). At each landmark *k* we construct a region covariance descriptor [13]. Specifically, for each patch $\mathcal{P}_{k,t}$ centered at *k* at time *t* we compute two covariance matrices, $\Sigma_{\mathbf{v}}$ and $\Sigma_{\mathbf{t}}$: The feature vector for landmark *k* is then $\mathbf{f}_k = [\boldsymbol{\sigma}_{\mathbf{v}}, \boldsymbol{\sigma}_{\mathbf{t}}]^\top$, where $\boldsymbol{\sigma}_{\mathbf{v}}$ is a 1×6 vector of upper-right entries of $\Sigma_{\mathbf{v}}$ and $\boldsymbol{\sigma}_{\mathbf{t}}$ is a 1×10 vector of upper-right entries of $\Sigma_{\mathbf{t}}$. We trained and tested our algorithm using the set of $\{\mathbf{f}_{k,t}\}$.

Fig. 2 illustrates several interesting qualitative results of our approach on the *Hot or Not* dataset. Table 1 provides a quantitative comparison between the accuracy of variants of our proposed approach, and the accuracy of seven different estimation methods proposed by [10] on the *Hot or Not* dataset. The first seven rows are results reported by [10]. The bottom four rows are variants of our method. We note that, unlike Glasner *et al.* [10], our "Kernel Est. with $w_\phi w_x$" method is capable of producing *time-varying* (dynamic) landmark weights (see Fig. 2), which provides a meaningful and intuitive way to understand which parts of the image contribute most significantly to the temperature estimate. At the same time, the accuracy of "Kernel Est. with $w_\phi w_x$" is competitive, which shows that our method does not sacrifice accuracy to achieve this capability.

4.2 Face age estimation

Face age estimation is a well-studied area in computer vision. For example, apparent age estimation [14] was a key topic in the 2015 Looking At People ICCV Challenge [15]. The top performers in that challenge all used a combination of deep convolutional neural networks and large training databases (*e.g.*, ~ 250*k* images). Given the significant engineering overhead required, we do not focus on achieving state-of-the-art accuracy using such large datasets. Instead, here we show qualitative results on a smaller age estimation dataset to illustrate several aspects of our approach. For experimentation, we used the Lifespan database [16], which has been previously used for age estimation [17] and modeling the evolution of facial landmark appearance [18].

The Lifespan database contains frontal face images with neutral and happy expressions, with ages ranging from 18 to 94 years. We used the 590 neutral expression faces with associated manually labeled landmarks from [17]. Following [17], we used five-fold cross-validation for our experiments.

Fig. 3 shows the age estimates and landmark weights produced by our method. We see that certain regions of the face (*e.g.*, eyes, mouth corners) generally received higher weights than others (*e.g.*, nose tip). However, this is not true for all faces. For example, cosmetics can alter appearance in ways that conceals apparent age, and landmarks can be occluded (*e.g.*, by hair or sunglasses). This implies that a globally consistent weight for each landmark is suboptimal. In contrast, our dynamic weights w_X adapt to each face instance to better handle such variations. See the supplementary material for additional results.

4.3 Alzheimer's disease (AD) classification

We further demonstrate the performance of our model on a clinically-relevant task of predicting disease status from neuroimaging data. For this set of experiments we used diffusion tensor imaging (DTI) data from an Alzheimer's disease (AD) dataset. We use the fractional anisotropy (FA) maps that are the normalized standard deviation maps of the eigenvalues of the DTI as a single channel image for deriving the feature vectors. We used standard image processing of DTI [19] to derive these measures in the entire white matter region of the brain from a total of 102 subjects. There were 44 subjects with AD diagnosis and 58 matched normal control (CN) subjects. We defined 186 regularly-placed landmarks on the lattice of the brain volume. At each of these landmarks we derived mean feature vector ([I, Ix, Iy, Iz]) using a local 3D patch of size 10 × 10 × 10. I is the FA value, Ix, Iy, Iz are the differentiated FA values in the x, y and z directions, respectively. Since our algorithm performs regression, we used {0, 2} for {CN, AD} and thresholded the prediction results at 1. Using these features we obtained a classification accuracy of 86.17% using 10-fold cross-validation. Even though our method is a regression model, this outperforms SVM with PCA on the same data set showing 80%–85% [20]. The resulting conditional confidence maps (computed using $w_X j$ in Alg. 1) for the top 20 landmarks (of the 186) for two sample subjects are shown in Fig. 4.

5 Conclusions

This paper provided a statistical algorithm for identifying conditional confidence of covariates in a regression setting. We utilized the concept of Sufficient Reduction within an Inverse Regression (AIR) model to obtain formulations that offer individual-level relevance of covariates. On all three applications described here, we found that in addition to gross accuracy, the ability to explain a prediction for each test example can be valuable for many applications. Our approach comes with various properties such as optimal weights, unbiasedness, and procedures to calculate conditional densities along only relevant dimensions given a regression task; these are interesting side results. Our evaluations suggest that there is substantial value in further exploring how Abundant Inverse Regression can complement current regression approaches in computer vision, offer a viable tool for interpretation/feedback, and guide the design of new methods that exploit these conditional confidence capabilities directly.

Acknowledgments

This research was supported by NIH grants AG040396, and NSF CAREER award 1252725. Partial support was provided by UW ADRC AG033514, UW ICTR 1UL1RR025011, UW CPCP AI117924 and Waisman Core Grant P30 HD003352-45.

References

1. Friston KJ, Holmes AP, Worsley KJ, Poline JP, Frith CD, Frackowiak RS. Statistical parametric maps in functional imaging: a general linear approach. Human Brain Mapping. 1994; 2(4):189–210.
2. Cao X, Wei Y, Wen F, Sun J. Face alignment by explicit shape regression. International Journal of Computer Vision. 2014; 107(2):177–190.
3. Li KC. Sliced inverse regression for dimension reduction. Journal of the American Statistical Association. 1991; 86(414):316–327.
4. Loh PL, Wainwright MJ. High-dimensional regression with noisy and missing data: Provable guarantees with non-convexity. Proceedings of the Advances in Neural Information Processing Systems. 2011:2726–2734.
5. Chen, Y.; Caramanis, C. Noisy and missing data regression: Distribution-oblivious support recovery; Proceedings of the 30th International Conference on Machine Learning; 2013. p. 383-391.
6. Krutchkoff R. Classical and inverse regression methods of calibration. Technometrics. 1967; 9(3): 425–439.
7. Taddy M. Multinomial inverse regression for text analysis. Journal of the American Statistical Association. 2013; 108(503):755–770.
8. Rabinovich, M.; Blei, D. The inverse regression topic model; Proceedings of the 31st International Conference on Machine Learning; 2014. p. 199-207.
9. Kim M, Pavlovic V. Dimensionality reduction using covariance operator inverse regression. Proceedings of the Computer Vision and Pattern Recognition, IEEE. 2008:1–8.
10. Glasner, D.; Fua, P.; Zickler, T.; Zelnik-Manor, L. Hot or not: Exploring correlations between appearance and temperature; Proceedings of the IEEE International Conference on Computer Vision; 2015. p. 3997-4005.
11. Adragni KP, Cook RD. Sufficient dimension reduction and prediction in regression. Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences. 2009; 367(1906):4385–4405.
12. Cook RD. Fisher lecture: Dimension reduction in regression. Statistical Science. 2007:1–26.
13. Tuzel, O.; Porikli, F.; Meer, P. Region covariance: A fast descriptor for detection and classification; Proceedings of the 9th European Conference on Computer Vision; 2006. p. 589-600.
14. Zhu, Y.; Li, Y.; Mu, G.; Guo, G. A study on apparent age estimation; Proceedings of the IEEE International Conference on Computer Vision Workshops; 2015. p. 25-31.
15. Escalera S, Fabian J, Pardo P, Baró X, Gonź J. Chalearn looking at people 2015: Apparent age and cultural event recognition datasets and results. Proceedings of the IEEE International Conference on Computer Vision Workshop. 2009; 367:4385–4405.
16. Minear M, Park D. A lifespan database of adult facial stimuli. Behavior Research Methods. 2004; 36(4):630–633.
17. Guo, GD.; Wang, X. A study on human age estimation under facial expression changes; IEEE Conference on Computer Vision and Pattern Recognition; 2012.
18. Kim, HJ.; Xu, J.; Vemuri, BC.; Singh, V. Manifold-valued Dirichlet processes; Proceedings of the International Conference on Machine Learning; 2015. p. 1199-1208.
19. Cook, P.; Bai, Y.; Nedjati-Gilani, S.; Seunarine, K.; Hall, M.; Parker, G.; Alexander, D. Camino: open-source diffusion-MRI reconstruction and processing. Proceedings of the 14th Scientific

Meeting of the International Society for Magnetic Resonance in Medicine; Seattle WA, USA. 2006.

20. Hwang, SJ.; Collins, MD.; Ravi, SN.; Ithapu, VK.; Adluru, N.; Johnson, SC.; Singh, V. A projection free method for generalized eigenvalue problem with a nonsmooth regularizer; Proceedings of the IEEE International Conference on Computer Vision; 2015. p. 1841-1849.

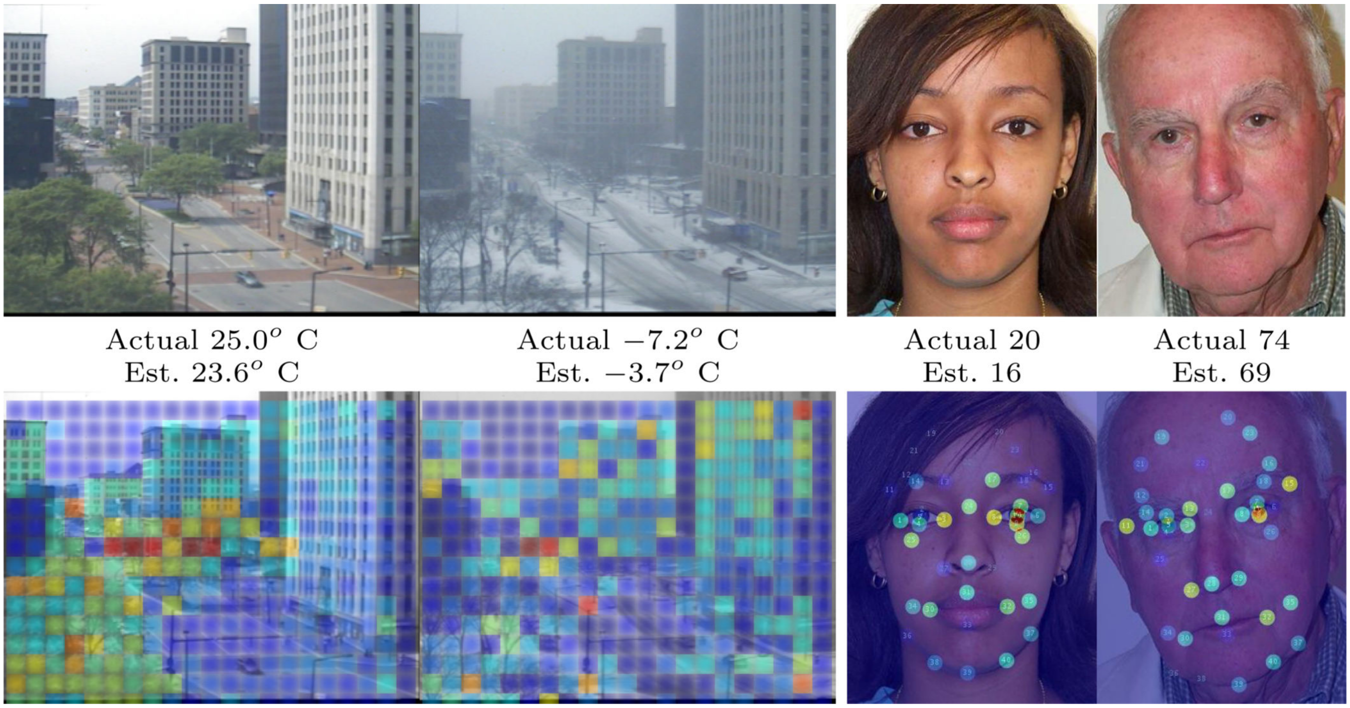

Fig. 1.

Dynamic feature weights for two tasks: ambient temperature prediction (left) and age estimation (right). Our formulation provides a way to determine, at test time, which features are most important to the prediction. Our results are competitive, which demonstrates that we achieve this capability without sacrificing accuracy.

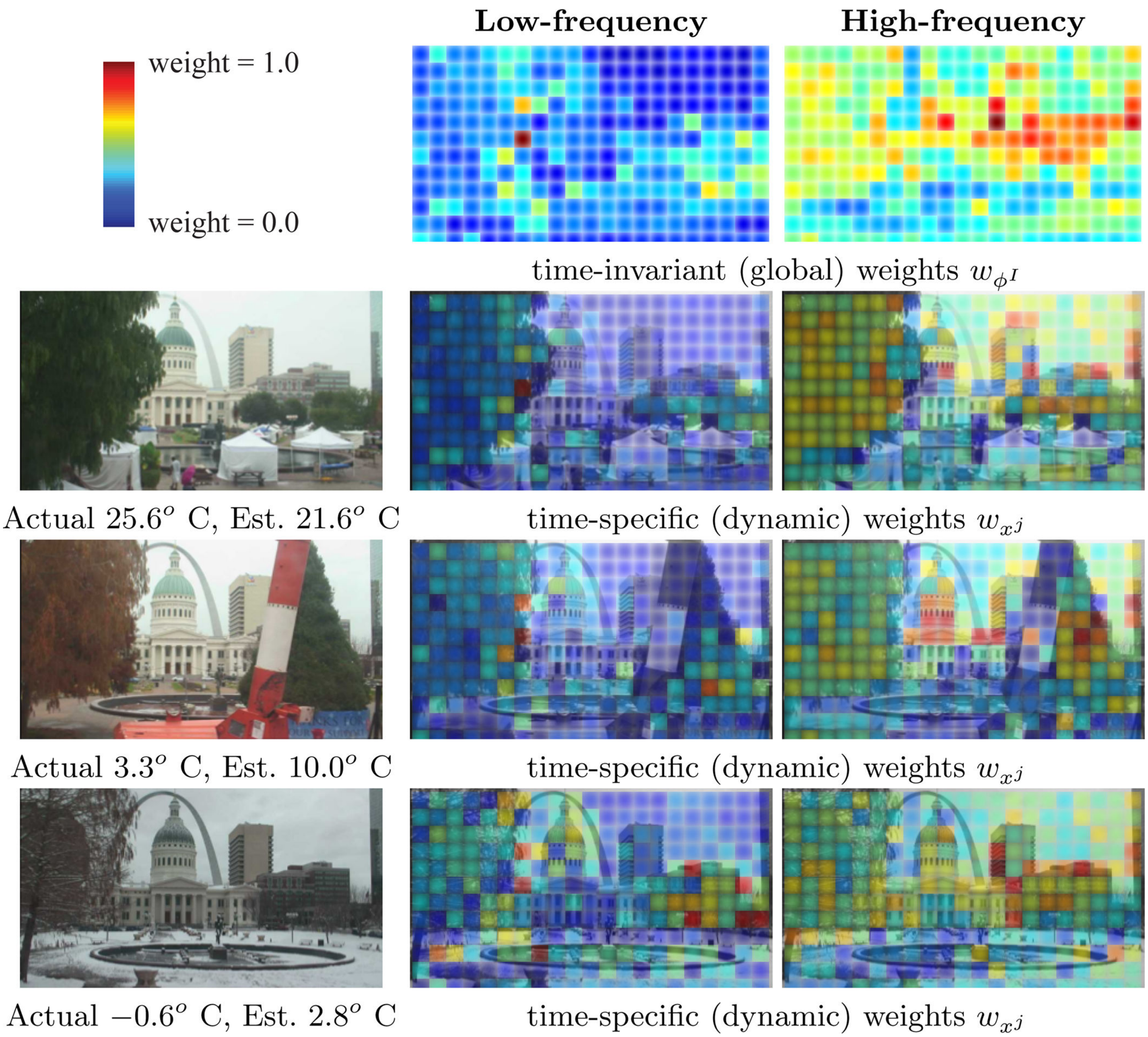
Low-frequency
High-frequency
weight = 1.0
weight = 0.0
time-invariant (global) weights w_{ϕ^I}
Actual 25.6^o C, Est. 21.6^o C
time-specific (dynamic) weights w_{x^j}
Actual 3.3^o C, Est. 10.0^o C
time-specific (dynamic) weights w_{x^j}
Actual -0.6^o C, Est. 2.8^o C
time-specific (dynamic) weights w_{x^j}

Fig. 2.

Qualitative results on scene (a) from the *Hot or Not* dataset [10]: summer, late autumn, and winter. Notice that low-frequency global weights (row 1, middle column) tend to be larger around the background trees and at the edge of the foreground tree, which reflects that leaf appearance is well correlated with the season. Observe that high-frequency global weights (row 1, right column) tend to be larger on distant buildings, which reflects the intuition that daily weather variations (*e.g.*, fog, precipitation) can dramatically change the appearance of the atmosphere, which is especially noticeable against the backdrop of distant buildings. Note that our method correctly reduces the high-frequency weights on the crane (row 3, right column), which suggests that unpredictably occluded landmarks should not contribute to the estimate (appearance temporarily becomes uncorrelated with temperature). **Best viewed in color**.

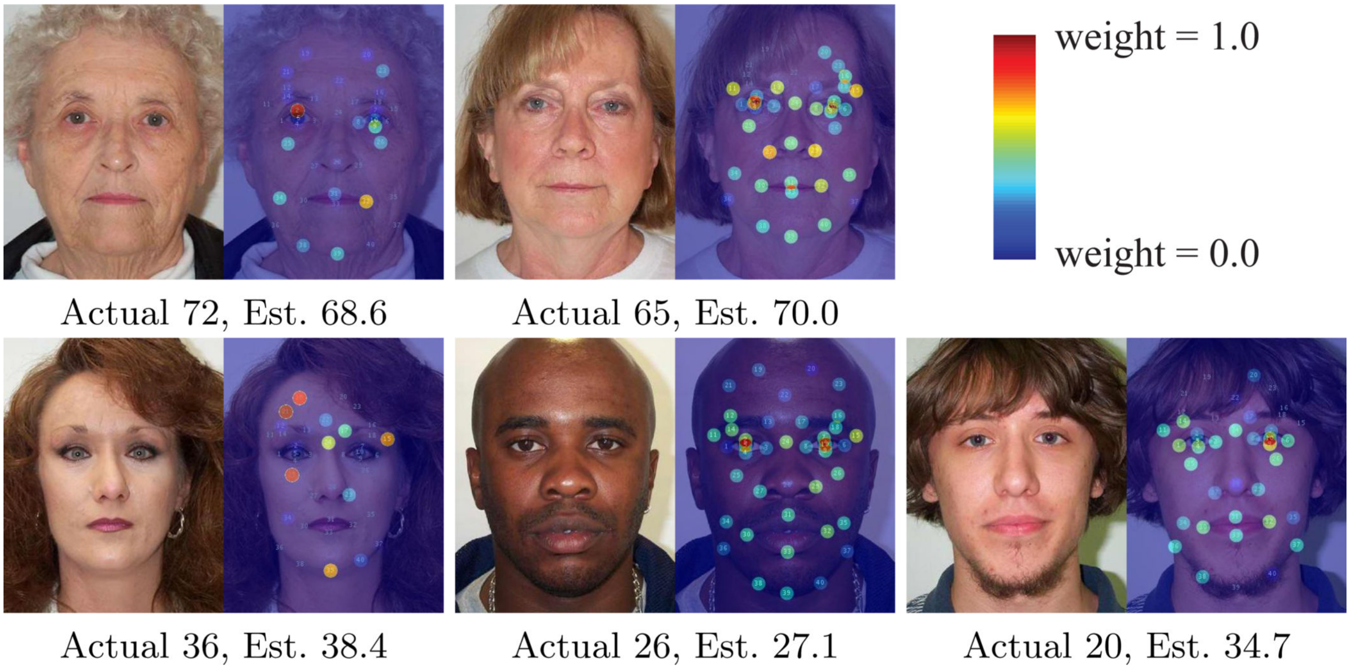

Fig. 3.

Qualitative age estimation results on images from the *Lifespan* database [16]. Notice that landmarks occluded by hair are correctly down-weighted. Eye and mouth landmarks tend to have higher weight, which suggests that their appearance is more predictive of age than the nose, for example. However, we see that the eye and mouth corners of the 36-year-old woman (second row, first column) are very low, perhaps due to her cosmetics. Our method is not always accurate. For example, the age estimate for the 20-year-old man (second row, third column) is technically incorrect. However, his apparent age is arguably closer to the estimate than his actual age. See the supplementary material for additional results. **Best viewed electronically in color**.

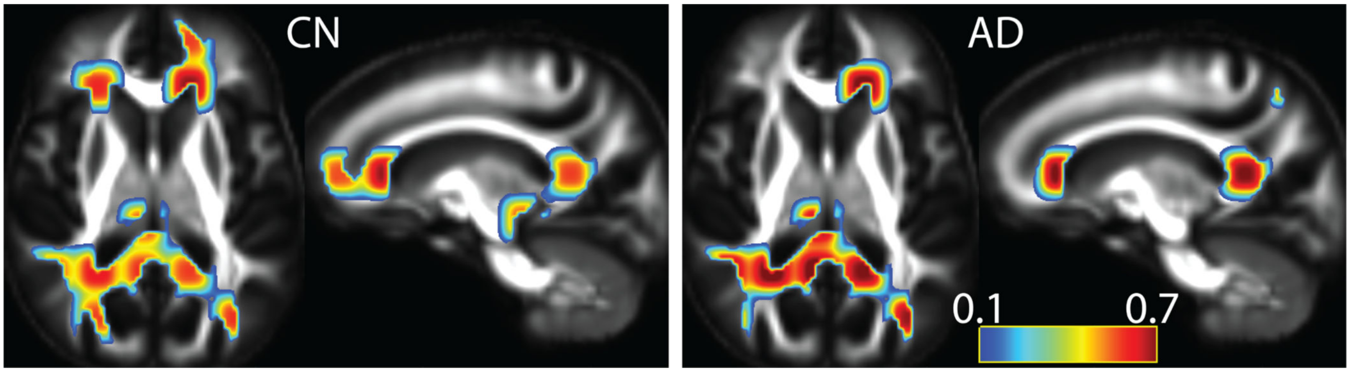

Fig. 4.

Conditional confidence maps of two representative subjects from the normal control group (left) and the AD group (right). The maps are overlaid on the population mean FA map. Observe that different white matter regions play important roles in the prediction. For example, the frontal white matter is bilaterally important in the CN subject where as there is assymetry in the AD subject.

Table 1

Accuracy of Celsius temperature prediction on *Hot or Not* [10]. Each cell contains two values: R^2 / RMSE, where $R^2=1-\frac{\mathrm{MSE}}{\sigma^2}$, MSE is the mean squared error of the temperature estimation, σ^2 is temperature variance, and RMSE is root MSE. The first seven rows are results from [10]. The bottom four rows are variants of our method. Our method produces a *time-varying* (dynamic) weight for each landmark, which provides a richer, more intuitive explanation of the estimation process.

	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)	(j)
Last Year	0.42 / 9.14	0.56 / 8.16	0.54 / 7.53	0.41 / 5.44	0.61 / 7.35	0.00 / 4.30	0.67 / 6.20	0.59 / 6.77	0.00 / 4.84	0.61 / 7.64
Nearest Neighbor Image	0.47 / 8.72	0.59 / 7.83	0.51 / 7.73	0.15 / 6.51	0.13 /10.92	0.00 / 4.57	0.16 / 9.89	0.70 / 5.83	0.00 / 4.44	0.62 / 7.47
Local Regression	**0.67 / 6.85**	0.65 / 7.24	0.70 / 6.03	0.59 / 4.53	0.76 / 5.77	0.38 / 3.19	0.50 / 7.63	0.77 / 5.09	0.10 / 3.68	0.59 / 7.77
LR Temporal Window	0.61 / 7.52	0.69 / 6.86	0.72 / 5.82	**0.64 / 4.23**	0.79 / 5.39	**0.53 / 2.77**	0.54 / 7.35	0.76 / 5.22	0.11 / 3.67	0.58 / 7.85
Global Ridge Regression	0.00 /18.16	0.78 / 5.74	0.00 /35.02	0.00 /11.37	0.00 /43.51	0.10 / 3.84	**0.74 / 5.54**	0.00 /13.86	0.23 / 3.41	0.46 / 8.91
Convolutional NN	0.49 / 8.55	0.79 / 5.59	0.71 / 5.96	0.24 / 6.17	0.61 / 7.36	0.48 / 2.90	0.39 / 8.48	**0.79 / 4.88**	0.43 / 2.93	0.66 / 7.12
Transient Attributes	0.36 / 9.60	0.70 / 6.69	0.58 / 7.20	0.55 / 4.75	0.68 / 6.62	0.21 / 3.59	0.58 / 7.03	0.65 / 6.31	0.16 / 3.56	0.67 / 7.00
Weighted Avg. with w_ϕ	0.54 / 8.13	0.66 / 7.18	0.00 / 13.00	0.38 / 5.59	0.69 / 6.54	0.35 / 3.26	0.49 / 7.74	0.12 / 9.96	0.34 / 3.17	0.58 / 7.91
Kernel Est. (no weights)	0.55 / 8.01	0.81 / 5.38	0.75 / 5.54	0.56 / 4.69	0.82 / 4.92	0.00 / 4.23	0.33 / 8.89	0.71 / 5.68	0.45 / 2.88	**0.72 / 6.49**
Kernel Est. with w_ϕ	0.13 / 11.15	0.81 / 5.32	0.74 / 5.59	0.41 / 5.43	**0.83 / 4.82**	0.20 / 3.62	0.39 / 8.42	0.71 / 5.68	0.53 / 2.67	0.68 / 6.93
Kernel Est. with $w_\phi w_x$	0.28 / 10.16	**0.81 / 5.30**	**0.76 / 5.41**	0.32 / 5.82	0.83 / 4.87	0.22 / 3.56	0.38 / 8.52	0.72 / 5.59	**0.55 / 2.62**	0.68 / 6.93