

Engineering with Computers

Data-driven Modeling of the Mechanical Behavior of Anisotropic Soft Biological Tissue --Manuscript Draft--

Manuscript Number:	EWCO-D-22-00085R1	
Full Title:	Data-driven Modeling of the Mechanical Behavior of Anisotropic Soft Biological Tissue	
Article Type:	Original Article	
Corresponding Author:	Adrian Buganza Tepole Purdue University West Lafayette, IN UNITED STATES	
Corresponding Author Secondary Information:		
Corresponding Author's Institution:	Purdue University	
Corresponding Author's Secondary Institution:		
First Author:	Vahidullah Tac, MS	
First Author Secondary Information:		
Order of Authors:	Vahidullah Tac, MS	
	Vivek Sree, MS	
	Manuel Rausch, PhD	
	Adrian Buganza Tepole	
Order of Authors Secondary Information:		
Funding Information:	Division of Civil, Mechanical and Manufacturing Innovation (NSF CMMI 1916668)	Dr. Adrian Buganza Tepole
	National Institute of Arthritis and Musculoskeletal and Skin Diseases (R01AR074525)	Dr. Adrian Buganza Tepole
	Division of Civil, Mechanical and Manufacturing Innovation (1916663)	Dr. Manuel Rausch
	Division of Civil, Mechanical and Manufacturing Innovation (2127925)	Dr. Manuel Rausch
Abstract:	Closed-form constitutive models are currently the standard approach for describing soft tissues' mechanical behavior. However, there are inherent pitfalls to this approach. For example, explicit functional forms can lead to poor fits, non-uniqueness of those fits, and exaggerated sensitivity to parameters. Here we overcome some of these problems by designing deep neural networks (DNN) to replace such explicit expert models. One challenge of using DNNs in this context is the enforcement of stress-objectivity. We meet this challenge by training our DNN to predict the strain energy and its derivatives from (pseudo)-invariants. Thereby, we can also enforce polyconvexity through physics-informed constraints on the strain-energy and its derivatives in the loss function. Direct prediction of both energy and derivative functions also enables the computation of the elasticity tensor needed for a finite element implementation. Then, we showcase the DNN's ability by learning the anisotropic mechanical behavior of porcine and murine skin from biaxial test data. Through this example, we find that a multi-fidelity scheme that combines high fidelity experimental data with a low fidelity analytical approximation yields the best performance. Finally, we conduct finite element simulations of tissue expansion using our DNN model to illustrate the potential of data-driven approaches such as ours in medical device design. Also, we expect that the open data and software stemming from this work will broaden the use of data-driven constitutive models in soft tissue mechanics.	

Response to Reviewers:

We would like to thank the Editor and reviewers for taking the time to read the original manuscript in detail and providing excellent and helpful comments. We have done a significant amount of work to address all concerns. The detailed response to the Reviewers addressing the concerns point by point is submitted with the revised manuscript and all changes marked in red in the revision. We hope that the revised manuscript will be acceptable for publication. Thank you again.

First of all we would like to thank the Editor and reviewers for the critiques of our manuscript, which we found very helpful during the revision. We have done an extensive amount of work to address the comments. In particular, we conducted a couple of studies in the Supplement to determine the optimal neural network architecture and hyperparameters. We changed the framework to enforce polyconvexity. The upgrades in the model show better performance with respect to the original submission. We have also improved the discussion significantly. We are hopeful that the revised manuscript will be suitable for publication in *Engineering with Computers*. Thank you again for the time to read the original manuscript and provide feedback.

Reviewer #1: This is very nice work on DNN based constitutive functions for soft tissue. I appreciate the many aspects included in this work: physics constraints such as convexity of the strain energy, matching energies and stresses, multi-fidelity models as well as data-augmentation. This work will serve as a nice primer for continued use of DNNs as replacements for "expert" constitutive functions. I have a few minor questions, comments and suggestions.

Thank you for the comments and feedback. We appreciate the review and have improved the manuscript accordingly. Please see below the point-by-point reply to the comments.

1. It would be helpful for the authors to expand on the remark (Page 7) that incompressibility can be exactly enforced. Since the remark appears early in the paper, perhaps it can be clarified in the Conclusions.

which relates derivatives with respect to the isochoric part of the deformation to derivatives with respect to the total deformation. If the material under consideration is exactly incompressible, i.e. $J = 1$, then $\hat{\mathbf{C}} = \mathbf{C}$, (5) reduces to

$$\mathbf{S} = 2[\hat{\Psi}_1 \mathbf{I} + \hat{\Psi}_2(\hat{I}_1 \mathbf{I} - \mathbf{C}) + \hat{\Psi}_{4a} \mathbf{V}_0 + \hat{\Psi}_{4s} \mathbf{W}_0] + p \mathbf{C}^{-1}, \quad (8)$$

Thank you for the comment. We have clarified right after eq. 8 that “and the pressure p becomes an unknown Lagrange multiplier field. In certain cases, p can be solved from boundary conditions. The nearly incompressible formulation is used in the finite element formulation. However, the incompressible formulation is used during neural network training since this constraint can be easily enforced for the plane stress biaxial deformations considered”.

2. With regard to equations (17) and (18), isn't the loss L_2 effectively imposing the same conditions as the second term of L_1 ?

$$\mathcal{L}_1 = \frac{1}{N} \sum_{n=1}^N \left[\left(\left(\Psi_{\text{iso}}^{\text{p}} \right)^{(n)} - \left(\Psi_{\text{iso}}^{\text{d}} \right)^{(n)} \right)^2 + \sum_{i=1,2,4v,4w} \left(\left(\hat{\Psi}_i^{\text{bp}} \right)^{(n)} - \left(\hat{\Psi}_i^{\text{p}} \right)^{(n)} \right)^2 \right], \quad (17)$$

$$\mathcal{L}_2 = \frac{1}{N} \sum_{n=1}^N \left\| \left(\sigma^{\text{p}} \right)^{(n)} - \left(\sigma^{\text{d}} \right)^{(n)} \right\|_F \quad (18)$$

We have improved the explanation in the text. We move the explanation of the second term of eq. 17 right after eq. 17, and modified it for clarity. The DNN predicts the strain energy and its derivatives with respect to the invariants, then those derivatives are post-processed to obtain stresses. \mathcal{L}_2 compares the stress post-processed from the energy derivatives to the stress data in the training set, whereas the second term of \mathcal{L}_1 compares the predicted derivatives (which are direct outputs of the neural network) are consistent with the derivatives obtained via back-propagation of the predicted strain energy. Thus in the second term of \mathcal{L}_1 no training data is involved. The text right after eq. 17 now reads:

The second term in eq. (17) is a regularization term that is added to the loss function to enforce that the predicted derivatives are consistent with the strain energy. In other words, the derivatives that are a direct output of the neural network, Ψ_i^{p} , should coincide with the derivatives of $\Psi_{\text{iso}}^{\text{p}}$ using back-propagation, denoted as Ψ_i^{bp} in eq. (17), with $i=1,2,4v,4w$, where $(*)^{\text{bp}}$ stands for "back-propagation".

3. The use of convexity as a physical constraint can be problematic, given the theoretical questions surrounding its appropriateness for soft materials. It may make the response too limited, and in particular may prevent the development of softer modes, and bifurcations--where they may exist physically. The later errors (Fig 4) associated with this constraint in the DNN may also reflect either noise or that this constraint is too restrictive. The effect of noise has been reported in a different data-driven learning setting in <https://doi.org/10.1016/j.jmps.2021.104474>. More discussion is perhaps warranted here.

Thank you for pointing this out. There are several points here. On the one hand we agree that convexity with respect to C might be too restrictive. We have changed to polyconvexity. In fact in the original paper we had a mistake since we were requiring convexity with respect to the invariants which is actually closer to polyconvexity than to convexity with respect to C . This is because I_2 is convex with respect to $\text{cof}(F)$ but it is not convex with respect to C . Thus, convexity with respect to I_2 can retain convexity with respect to $\text{cof}(F)$ but not necessarily with respect to C . For polyconvexity, however, we have the additional requirement that the function has to be non-decreasing with respect to the invariants. This is because I_1 , for example, is linear in C and thus convexity with respect to I_1 was enough to retain convexity with respect to C . I_1 , on the other hand, is quadratic with respect to F . Thus, to retain convexity with respect to F , the strain energy has to be convex and non-decreasing with respect to I_1 . Similar arguments follow for I_{4v} , I_{4w} , and I_2 , which are quadratic with respect to F or $\text{cof}(F)$. In

summary, we have corrected the text and the method to impose polyconvexity by requiring convexity with respect to the invariants and non-negative derivatives. In the text this is explained in the Methods:

To guarantee existence of a solution to the boundary value problem, a suitable constraint on the strain energy is that of polyconvexity with respect to the deformation gradient, F . An alternative approach is to enforce convexity of the strain energy with respect to C . Convexity with respect to C can lead to the existence of global minima in boundary value problems under certain conditions. This convexity condition has been employed in constitutive modeling of biological tissues and in numerous studies on data-driven models of hyperelastic materials. For instance, the popular constitutive model by Holzapfel, Gasser and Ogden to capture the mechanical behavior of collagenous tissues was developed to fulfill this condition. However, convexity with respect to C and polyconvexity with respect to F are not equivalent. Polyconvexity of the strain energy function with respect to F , together with some growth conditions on the strain energy, guarantees the existence of global minimizers to the total potential energy functional.

In the current study we enforce convexity of the strain energy function with respect to the deformation invariants of C with a global minimum at or below the point $I_1=3$, $I_2=3$, $I_{\{4v\}}=1$ and $I_{\{4w\}}=1$. The invariants are already convex functions of F or $\text{cof}(F)$. Thus, the non-decreasing convex function of the invariants, for $I_1 \geq 3$, $I_2 \geq 3$, $I_{\{4v\}} \geq 1$ and $I_{\{4w\}} \geq 1$, together with a suitable volumetric energy convex in J results in polyconvexity of the strain energy with respect to F .

However, the reviewer is right that maybe the DNN did not have enough flexibility to both fit the data and impose convexity. We have done an extensive hyperparameter study shown in the Supplement and we have used this study to define a better architecture and better weights on the losses. The new results are much better than the original, see Figure 4. Additionally, the effect of noise was considered in the Supplement to show that the neural network is still able to learn the mean response accurately. Thus, we do not think the problem is noise. Instead, we think that the problem with the experimental data is indeed related to lack of convexity, perhaps due to dissipative phenomena such as viscoelasticity or damage. We highlight this in the Discussion:

When trained on sparse high fidelity data alone, imposing convexity requirements ensures a physically admissible model, but comes at the expense of fitting error (see Figure 4). This result points to polyconvexity as a potentially restrictive condition on the data, possibly due to the existence of dissipative phenomena such as viscoelasticity or damage which were not accounted for in the model.

4. Are the hyperparameters a_0-a_3 determined by cross validation?

We have added a table and explanation for how we obtain the hyperparameters from a cross-validation study. The new data is in the Supplemental Material.

5. It would be useful to see learning curves--in the Appendix, perhaps?

We have added the additional training loss curves to the Supplementary Material. Figure 2 in the original submission already had the evolution of the loss for the synthetic data, but the supplement now shows the similar trends in the experimental datasets.

6. The caption of Fig 3 should mention what the colors mean.

We have done so.

7. More discussion of the poorer performance for training with off -biaxial data and testing against strip-biaxial data would be nice.

We have changed the figure completely since we changed the DNN architecture and added the non-negative restriction on the predicted derivatives. The new results do not show the undesirable dip in the predicted response by the DNN that was seen in the original figure (in which the stress was even negative). Now all the predictions show realistic curves. On the other hand, there are still differences in the validation errors depending on which curves are used for training. Errors in training are always low for the DNN, but validation errors do vary. We do not have a good explanation regarding which training data is preferred. It is clear from Figure 6 that the more data, the better the performance. On the other hand, from Figure 5 it is also clear that multi-fidelity data can allow the DNN to perform well in the regions with experimental data while approximating the GOH model elsewhere, as illustrated in Figure 5L especially.

8. How much slower is the DNN form of the strain energy versus the "expert" constitutive functions in the ABAQUS implementations?

This is an excellent question. It is about twice as slow to evaluate the NN model compared to the GOH model:

Execution time for GOH (Torsion): 00:01:42

Execution time for NN (Torsion): 00:02:24

We have added these times to the results.

It should also be noted that the DNN material model usually requires approximately twice as much computational time compared to the built-in GOH model. For example, for the torsion problem, the execution time for GOH is 00:01:42, while for the DNN UMAT it is 00:02:24.

Reviewer #2: The manuscript presents an interesting study to develop a deep neural network (DNN)-based constitutive model for modeling soft biological tissues -- anisotropic hyperelastic materials. Special customization is made to the DNN model to enforce physics constraints. Like other DNN constitutive approaches, this study adopts strain invariants as the DNN inputs and the strain energy as the outputs to guarantee objectivity, but it also set the derivatives of strain energy as direct DNN outputs. Consequently, the authors can use an alternative way (proposed in a previous paper Computer Methods in Applied Mechanics and Engineering Volume 387, 15 December 2021, 114160 by the same research group) to impose convexity constraint by enforcing the Hessian matrix of the energy derivatives to be positive definite, see Eqs. (22-23); meanwhile, an additional regularization (loss term) is introduced in the total loss function to guarantee the consistency between the direct derivatives and the backpropagation derivatives (Eq. (17)). Furthermore, the authors investigate the performance of the multi-fidelity scheme and the implementation of the proposed DNN model in Abaqus through a UMAT subroutine.

While the manuscript is well written and of great interest to Engineering with Computers readers, the novelties of the article seem not adequately explained and the numerical results are not clear enough to corroborate the proposed method. I have a few concerns/comments listed as follows that may improve the manuscript:

General comments:

1. Considering there have been quite a few DNN-based constitutive models, see the papers reviewed in (Klein et al. 2022; Kalina et al. 202), it is not very clear to the reviewer that how the proposed method distinguishes from the state-of-art, especially compared to the DNN models that also consider imposing convexity (Klein et al. 2022; Kalina et al. 2022; Heider et al. 2020). Also, it would be beneficial for readers if the manuscript provides review and comment on the model-free approaches in biomechanics, such as (Mora-macías et al. 2020; He et al. 2021a; He et al. 2021b). Overall, it is suggested more discussion about the advantages and advantages of the present study against the previous research.

Thank you for pointing out these references which have enhanced our discussion significantly. Since we changed the discussion in many parts, we do not copy all the changes here. Rather, we have marked all the changes in red in the revised version, but we would like to summarize here the main points:

- Use of experimental data from murine and porcine skin (11 mice and 2 pigs). This is in contrast to other work in the literature, which has often considered synthetic datasets
- Prediction of stress tensor and consistent tangent and satisfaction of objectivity, all of which are necessary for the finite element implementation. This is in contrast with methods that rely on approximating the stress directly, or with methods that predict the energy and are thus not generally showcased in realistic finite element applications
- Convexity and objectivity are certainly a major focus in the data-driven model literature. Our approach is slightly different to the papers suggested by the

reviewer, but along the same lines. Klein et al. 2022 imposes polyconvexity through convex activation functions and non-negative weights while we do it through the loss, they also use the invariants as inputs. Kalina et al. 2022 also uses invariants as inputs but predicts principal stresses. This can only be done for isotropic materials and even then, the solution is not always unique and that's why the authors regularize by integrating the energy and adding a loss that enforces the Clausius-Duhem inequality. Heider et al. impose rotational invariance through an additional loss function. Overall these are all effective ways of dealing with issues of objectivity and convexity and we have slightly different strategies but with the same goal. We have thus highlighted these efforts in the discussion.

Regarding the model-free methods. This is a great comment. We have now added this in the Discussion regarding model-free methods:

When enough data is available, recent efforts in data-driven computational mechanics have shown that model-free approaches can be used (Kirchdoerfer and Ortiz 2016, Mora et al. 2020). However, for applications in biomechanics, the anisotropy and high nonlinearity in the materials necessitates large amounts of data from deformations that can cover the entire input space (He et al. 2021a, He et al. 2021b). This is often out of reach soft tissue characterization. Thus, we propose a DNN model that captures the experimental data, but does so constrained by a hyperelastic framework and the condition of polyconvexity of the strain energy.

2. In this study, both the (isochoric) strain energy and its 4 derivatives are taken as DNN outputs. The measured data associated with isochoric strain energy and stresses are used for training, see Eq. (17) and (18). However, how to collect (isochoric) strain energy data in real experiments of biological tissues? For example, does the three experimental datasets (in Page 12) contain any strain energy data?

This is a good point. The reviewer is correct in that the contribution of isochoric and volumetric energies would require different experiments and assumptions. However the incompressibility and plane stress conditions for biaxial testing of thin biological tissues is what enables us to integrate the strain energy from the stress-strain curves and associate it with the isochoric deformation. We have clarified in the methods: "Training data for the DNN is in the form of stretch and stress data, as well as the values of strain energy obtained by the integration of the stretch-stress curves."

3. Following the above question, because both the derivatives of the strain energy and the Hessian matrix in (21) can be computed simply by using standard first and second-order automatic differentiation (i.e., backpropagation), what is the advantage of the proposed model that considers the 4 derivatives as direct DNN outputs? More DNN outputs require larger DNN model, leading to increasing training effort. In addition, is there any normalization scheme used in the output layer, which could be important for multiple outputs?

This is a good point that we discuss in the revised text. The reviewer is right in that automatic differentiation can be used to obtain the second derivatives. However, this approach can introduce noise and error in the second derivatives, as pointed out for example by Vlassis et al. 2020, where the authors build DNN to learn the energy but then need to regularize the derivatives to match the stress data and do not consider second derivatives. Consider the case of ReLu for instance. Even though it is able to capture energies quite nicely, the second derivatives are zero and thus ReLu activation functions cannot be used if accurate second derivatives are needed. In our approach, since energy and derivatives are predicted separately, one could use ReLu functions and still be able to obtain second derivatives. This is akin to multifield problems in elasticity, or enhanced strain elements. The addition of the first derivatives as degrees of freedom actually provides a lot of flexibility and emphasizes our priority which is to fit the stress data. In fact, one could ignore the strain energy completely. In turn, a compatibility condition to guarantee that the derivative functions do indeed come from differentiation of an underlying energy can be imposed without reconstructing the energy itself. One of the papers suggested by the Reviewer does something similar. In Kalina et al. 2022, the authors learn a map between invariants and principal stresses. However, the principal stresses are not always uniquely defined. The authors integrate the energy in order to regularize their stress predictions. Alternatives to deal with these problems are integrable neural networks. In recent work we have explored the use of neural ordinary differential equations to construct polyconvex material models, and in that case we completely ignore the energy and predict energy derivatives only. The architecture we use in that case guarantees the compatibility of the derivative functions. We have changed the discussion:

The DNN design shown here, including the use of invariants as inputs and prediction of energy and energy derivatives as outputs, allows us to compute not only the stresses but the consistent tangent. Together with the polyconvexity loss, our data-driven framework is uniquely suited for finite element simulations. While it would be possible to predict the energy alone, this introduces noise in the derivatives that need to be regularized as shown in (Vlassis et al 2020). An alternative framework is to use integrable neural networks (Teichert et al 2019). Another method we have explored recently is the use of neural ordinary differential equations to learn the energy derivatives, ignoring the underlying energy function entirely (Tac et al 2021). In (Tac et al 2021), the model architecture guarantees that the derivative functions do indeed come from differentiation of an underlying potential even if this potential is not explicitly modeled. The approach followed here is more akin to multifield formulations in elasticity or enhanced strain methods, for which additional degrees of freedom are added together with suitable constraints (Wriggers et al 1996).

4. Page 8, Line 10-15: It states that the stresses in Eq. (18) are obtained by using the strain energy output, whereas Figure 1 shows that they are computed by the predicted energy derivates. Please clarify.

stress, defined in Eqs. (8) and (9), is computed based on the neural network outputs $\Psi_{\text{iso}}^{\text{p}}$, to produce σ^{p} . The loss can then be simply stated as

$$\mathcal{L}_2 = \frac{1}{N} \sum_{n=1}^N \left\| \left(\sigma^{\text{p}} \right)^{(n)} - \left(\sigma^{\text{d}} \right)^{(n)} \right\|_F \quad (18)$$

where $\|\cdot\|_F$ denotes the Frobenius norm

We have changed the text to reflect the correct approach. That is, we use the predicted energy derivatives to compute the stress and compare against the stress data. The energy is only used in the first term of the loss.

5. Page 9, Eq. (24): The weight coefficients will introduce bias in training the multi-loss functions, therefore it is critical to choose these coefficients. Could the authors share more insights about how to select those coefficients, and how is the sensitivity of prediction against them.

Excellent point. We added a hyperparameter optimization study to the supplement.

6. Page 11: While it is not a bad idea to start with the GOH model, the results in Figure 6 shows it is not a proper phenomenological model to fit the given experimental data because it doesn't consider the shear contributions, such as I_2. So, it could not be a fair comparison nor informative. Is the GOH selected on purpose to provide a low fidelity model? Could the authors explain why choose the GOH model?

The GOH is one of the most widely used material models for skin. We have experimented with other material models like Fung, Mooney Rivlin etc but GOH seems to perform best in almost all test cases. Which prompted us to choose this model as a benchmark. We have added a study in the Supplement comparing against other material models.

7. Page 10, Table 1: The DNN architecture is relatively small (only 2 hidden layers with 4~8 neurons per layer), which may lead to underfit the data. The reviewer suggests providing more analysis on the effect of DNN size. As shown in Figure 2-4, the fitting to the synthetic and experimental data are not satisfactory, and it could be improved by properly increasing the DNN size.

This is another excellent point. We also conducted additional studies changing the DNN architecture to pick the best one. As you can see in the revised figures in the main text and in the Supplement, the addition of an additional layer did provide an improvement

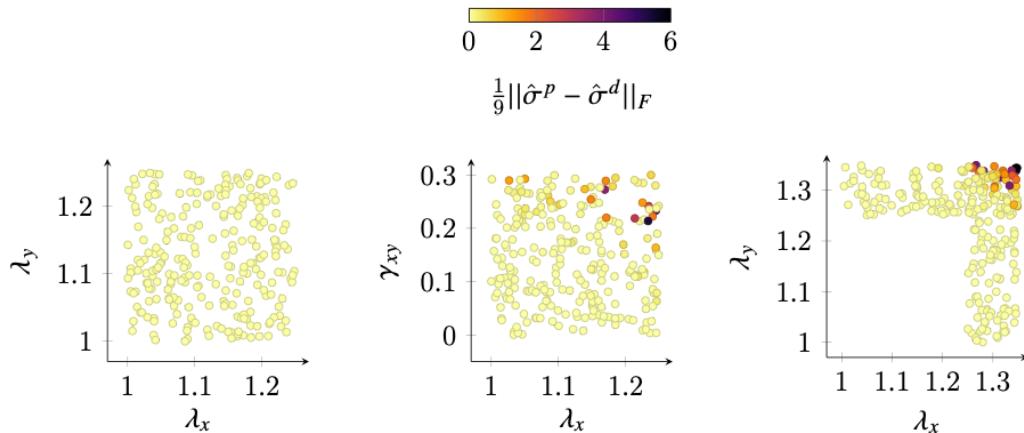
with respect to the original, while going beyond that did not improve the predictions further.

8. Page 14, Figure 2: Figure 2C shows loss-1, loss-2, and loss-3. How about loss-4? In Figure 2D-F, the differences between the predicted and ground truth curves are not clearly visualized, especially for the region with stretch close to 1. It would be better to zoom in or show their errors directly.

To increase the clarity of the paper we have renamed the losses and we now only have three losses.

9. Page 15, Figure 3: The relative error could be misleading. Would it more appropriate to show the max error of different stress components?

Definitely! We changed to absolute error but using the normalized stresses:



The stress $\hat{\sigma}^P$ and $\hat{\sigma}^d$ are normalized such that each entry of these tensors is obtained by subtracting the mean and dividing by the standard deviation of the stress values over the validation dataset, e.g. $\hat{\sigma}_{ij}^P = (\sigma_{ij}^P - \sigma_{ij}^{d,\text{avg}})/\Sigma_{ij}^d$, with $\sigma_{ij}^{d,\text{avg}}$ the mean of that stress component over the validation data, and Σ_{ij}^d the corresponding standard deviation. If the data were normal, then the normalized quantities would be almost entirely in the range $[-3, 3]$. Even though the data is not normal, this is a useful scaling of the stress. Relative errors are high in the low strain region for which the stress is negligible (orders of magnitude lower than in the high stress regions), and absolute errors are higher in regions of high stress even if the relative error is small. The normalized stress better captures the performance of the DNN over the input space. It can be seen that the DNN performs well within the training region but worse toward the boundary of the training region.

10. Page 17, Figure 4: As mentioned in Comment #7, the poor fitting performance, e.g. Figure 4F, could be due to the small size of DNN, which cannot well represent the variability of dataset. It is highly recommended to investigate the hyperparameters of DNN architecture.

We appreciate the comment and have done extensive amount of work shown in the Supplement to better tune the network architecture and the hyperparameters, with excellent results!

11. Page 17, Figure 4: Comparing C and G, it looks the imposition of convexity helps the DNN model better predict the extrapolative scenarios. However, the error values are 6.14 against 7.6, which indicates that the one with convexity constraint is worse. Could the authors explain?

This figure has changed significantly from the original version of the article. Now actually the polyconvexity constraint seems to increase the fitting errors. The non-negative derivatives obtained with the change to quadratic activation functions now prevents automatically some of the undesired behavior we had originally in which the predicted stresses would drop with higher deformations. The non-negative derivatives are an additional requirement of the formulation in addition to the convexity with respect to the invariants:

To guarantee existence of a solution to the boundary value problem, a suitable constraint on the strain energy is that of polyconvexity with respect to the deformation gradient, F . An alternative approach is to enforce convexity of the strain energy with respect to C . Convexity with respect to C can lead to the existence of global minima in boundary value problems under certain conditions. This convexity condition has been employed in constitutive modeling of biological tissues and in numerous studies on data-driven models of hyperelastic materials. For instance, the popular constitutive model by Holzapfel, Gasser and Ogden to capture the mechanical behavior of collagenous tissues was developed to fulfill this condition. However, convexity with respect to C , and polyconvexity with respect to F are not equivalent. Polyconvexity of the strain energy function with respect to F , together with some growth conditions on the strain energy, guarantees the existence of global minimizers to the total potential energy functional.

In the current study we enforce convexity of the strain energy function with respect to the deformation invariants of C with a global minimum at or below the point $I_1=3, I_2=3, I_{4v}=1$ and $I_{4w}=1$. The invariants are already convex functions of F or $\text{cof}(F)$. Thus, the non-decreasing convex function of the invariants, for $I_1 \geq 3, I_2 \geq 3, I_{4v} \geq 1$ and $I_{4w} \geq 1$, together with a suitable volumetric energy convex in J results in polyconvexity of the strain energy with respect to F .

But it now seems that imposing the polyconvexity constraint does not necessarily lead to better fits in the validation as shown in Figure 4C. Nevertheless, Figure 4D does show

that the convexity constraint is not satisfied in most of the input space. We think that enforcing the convexity is still desirable for the stable finite element implementation and due to the difficulty of the DNN models to extrapolate, as shown in Figure 3B,C. Even without imposing the convexity constraint directly, when using the analytical model as a low-fidelity approximation to help the DNN interpolate in regions without experimental data, the fact that the analytical model is polyconvex yields DNN fits that are mostly convex as seen in Figure 4L. Convexity constraints can help when there is noise in the data, as shown in the Supplement. We now state in the Discussion:

Imposing polyconvexity through the loss function ensures a stable material model suitable for finite element applications, but comes at the expense of fitting error (see Figure 4). This result points to polyconvexity as a potentially restrictive condition on the data, possibly due to the existence of dissipative phenomena such as viscoelasticity or damage which were not accounted for in the model. Noise can also affect the performance of the machine learning approaches (Wang et al 2021). The Supplement shows that imposing the convexity constraint, the DNN model can capture the synthetic data even in the presence of noise.

12. These questions may be out of scope of this paper and just for discussion. 1) As various phenomenological models may use different combination of invariances, could the authors share some comments on how to decide which invariances should be used as DNN inputs in the first place? 2) The study discusses about multifidelity scheme. But it is difficult to a priori estimate the uncertainty/errors associated with the low fidelity data, so how to properly select the coefficient a_3 for multi-fidelity seems not straightforward.

These are all great points. We have added a note in the Discussion noting the limitation:

Of course, this raises the question of how to balance between the two. Here we set a much higher priority for the experimental data, but future studies should quantify the uncertainty of both the data and the models in regions with little experimental data in order to rigorously weight the high and low fidelity models. With the current weights, we showcase the ability of the DNN material model to capture the mechanical response of skin based on data obtained from 2 pigs and 11 mice, demonstrating the applicability of our approach to realistic datasets.

Minor Comments:

- * Page 12: The weights between high and low-fidelity data are 10:1 but in Page 3 it states that $a_3=50$. Please clarify.

Typo, fixed

- * Page 14, Figure 2: The range in subfigure A is [0.8-1.3], which is inconsistent to [1.0, 1.25] indicated in Page 13 Line 17.

Typo, fixed

* Eq. (9), To be consistent, should the Cauchy stress symbol be bold?
It is bold but we are having a hard time making it look bolder. Hopefully, if accepted, the symbol would look bold in the font used by the journal.

* Page 16, Line 22: Should it be Figure 4L?

Typo, fixed

* Page 23 Line 54: The Figure reference is missing.

Fixed

[Click here to view linked References](#)10
11
12
13
14
15
16
17
18

19 Data-driven Modeling of the Mechanical 20 Behavior of Anisotropic Soft Biological Tissue 21 22

23
24 Vahidullah Tac¹, Vivek D. Sree¹, Manuel K. Rausch^{2,3,4}
25 and Adrian B. Tepole^{1,5*}
26

27 ¹School of Mechanical Engineering, Purdue University, West
28 Lafayette, IN, United States.

29 ²Department of Aerospace Engineering and Engineering
30 Mechanics, The University of Texas at Austin, Austin, TX,
31 United States.
32

33 ³Department of Biomedical Engineering, The University of Texas
34 at Austin, Austin, TX, United States.
35

36 ⁴Oden Institute for Computational Engineering and Sciences,
37 The University of Texas at Austin, Austin, TX, United States.
38

39 ⁵Weldon School of Biomedical Engineering, Purdue University,
40 West Lafayette, IN, United States.
41
42

43 *Corresponding author(s). E-mail(s): abuganza@purdue.edu;
44 Contributing authors: vtac@purdue.edu; vdharman@purdue.edu;
45 manuel.rausch@utexas.edu;
46
47

48 Abstract

49
50 Closed-form constitutive models are currently the standard approach for
51 describing soft tissues' mechanical behavior. However, there are inherent
52 pitfalls to this approach. For example, explicit functional forms can
53 lead to poor fits, non-uniqueness of those fits, and exaggerated sensitivity
54 to parameters. Here we overcome some of these problems by designing
55 deep neural networks (DNN) to replace such explicit expert models. One
56 challenge of using DNNs in this context is the enforcement of stress-
57 objectivity. We meet this challenge by training our DNN to predict the
58 strain energy and its derivatives from (pseudo)-invariants. Thereby, we
59 can also enforce polyconvexity through physics-informed constraints on
60 the strain-energy and its derivatives in the loss function. Direct prediction
61 of both energy and derivative functions also enables the computation
62 of the elasticity tensor needed for a finite element implementation. Then,
63
64

9
10
11
12 2 *Data-driven Soft Tissue Mechanics*
13

14 we showcase the DNN's ability by learning the anisotropic mechanical
15 behavior of porcine and murine skin from biaxial test data. Through this
16 example, we find that a multi-fidelity scheme that combines high fidelity
17 experimental data with a low fidelity analytical approximation yields
18 the best performance. Finally, we conduct finite element simulations of
19 tissue expansion using our DNN model to illustrate the potential of data-
20 driven approaches such as ours in medical device design. Also, we expect
21 that the open data and software stemming from this work will broaden
22 the use of data-driven constitutive models in soft tissue mechanics.

23
24 **Keywords:** Machine Learning , Nonlinear finite elements , Constitutive
25 modeling , Abaqus User Subroutine UMAT , multi-fidelity models , Skin
mechanics
26
27
28

29 1

Introduction

30

31 Skin is the largest organ in the body and understanding its mechanical prop-
32 erties is a crucial step in many biomedical applications, from prosthesis design
33 to surgical intervention [1]. The tissue microstructure is characterized by the
34 presence of semi-flexible biopolymer fiber networks such as collagen and elastin,
35 which endow skin with nonlinear and anisotropic behavior [2]. The mechani-
36 cal properties of skin are actually common across many soft connective tissues
37 [3, 4]. Traditionally, the mechanics of skin and other soft tissues has been mod-
38 elled using expert-constructed constitutive equations [5–7]. In this approach, a
39 closed-form expression describing the main features of the mechanics of a fam-
40 ily of materials is constructed first. Then, the free parameters in the equations
41 are fitted to a specific material in the family to obtain a calibrated model.
42 Inherent restrictions of the explicit functional form can result in poor fitting
43 and high sensitivity to parameters [8]. Unfortunately, even considering just
44 skin out of all connective soft tissue, there is currently no consensus on the
45 choice of model that is most suitable in a particular application [9–11].
46

47 A new, emergent approach to material modeling is the use of data-driven
48 methods [12, 13]. Among them, deep neural networks (DNN) have been suc-
49 cessfully employed to describe the mechanical behavior of several materials
50 [14–17]. In this approach, there is no need to limit the model to an analyti-
51 cal representation, which results in more accurate predictions than traditional
52 models [18–20]. Physics constraints such as objectivity of the stress and con-
53 vexitity of the hyperelastic strain energy potential are naturally satisfied by
54 most closed-form constitutive models [21]. These constraints are embedded
55 into data-driven methods either during the design of the algorithm itself [22],
56 or as a penalty [23]. Drawbacks of existing approaches stem from scarcity of
57 high fidelity test data to train the data-driven models [19]. This limitation is
58 particularly prevalent in soft tissue mechanics. Additionally, there is a lack of
59
60

data-driven material model software that can function in standard finite element solvers, which severely limits the applicability of these emerging methods to biomedical applications.

The route followed in our current work lies between the purely data-driven approach and the expert modeling approach. Closed-form material models already include knowledge of physics relevant to soft tissue, observations of the underlying microstructure, and intuition from the modeller regarding the main features of the material response. For example, to model skin, we have assumed hyperelasticity and used expert-designed strain energy functions to fit murine and porcine skin data [24]. However, the error in the fits can be undesirable, the parameters non-unique, and the predictions can be highly sensitive to the parameters [25]. Here we design DNN constitutive models and train them on multi-fidelity data: analytical strain energy functions serve as low fidelity approximations, high fidelity experimental measurements complement the data set. This approach is based on the recent literature that shows the advantage of multi-fidelity schemes over single fidelity approaches [16, 17, 26].

The proposed DNNs output the strain energy and its derivatives with respect to the isochoric strain invariants, including anisotropy, satisfying stress-objectivity *a priori*. The loss function is designed to impose polyconvexity of strain energy. The multi-output design in which both the energy and derivatives are predicted independently by the DNN, but coupled through additional loss terms, provides more flexibility during training and enables the computation of the stress and elasticity tensors. As a result, we are able to implement a DNN user material (UMAT) subroutine for the widely used non-linear finite element package Abaqus [27], and showcase its potential to impact skin therapeutics through simulations of tissue expansion. The work shown here will extend the reach of machine learning tools to improve the modeling of soft tissue mechanics, in particular through improved constitutive models ready to be used in commercial finite element codes.

2 Methods

Constitutive equations for a hyperelastic material with two families of fibers

In this study we use a Helmholtz free energy, Ψ , that is a function of the right Cauchy-Green deformation tensor, \mathbf{C} , and two material direction vectors in the reference configuration, \mathbf{v}_0 and \mathbf{w}_0 . This form of the Helmholtz free energy function allows for greater flexibility in recreating the mechanical behavior of materials where more than one family of fibers is present or even when the orientation of fibers is random, which is usually the case in biological tissues. For soft tissues, which we assume to be nearly incompressible, the additive split into isochoric and volumetric parts is used [5],

$$\Psi = \Psi_{\text{iso}}(\hat{I}_1, \hat{I}_2, \hat{I}_{4v}, \hat{I}_{4w}) + \Psi_{\text{vol}}(J), \quad (1)$$

4 Data-driven Soft Tissue Mechanics

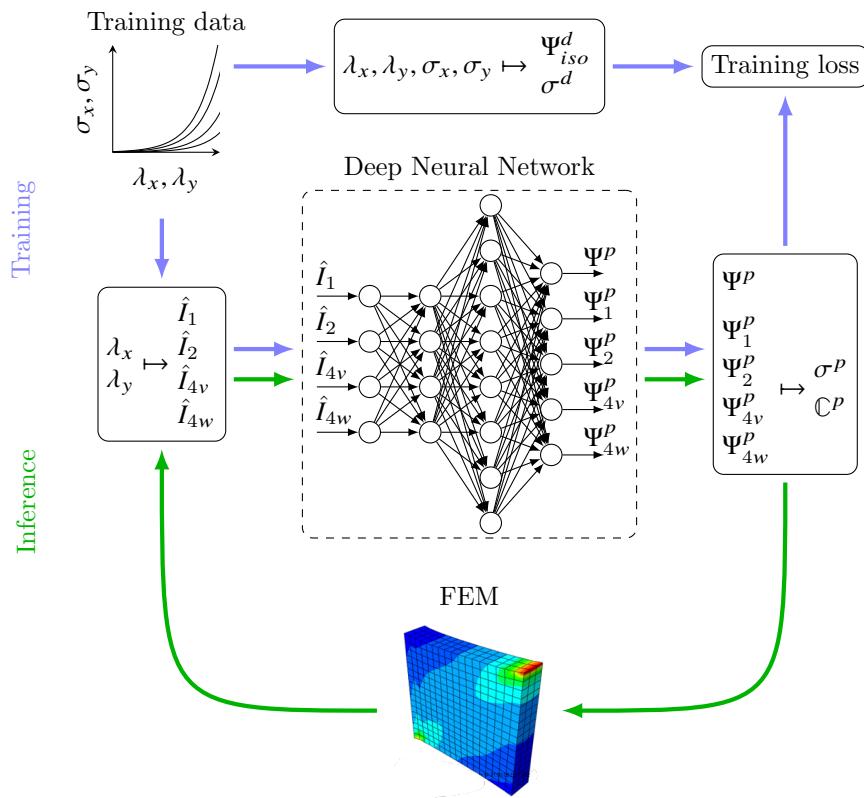


Fig. 1 Diagram depicting the training and inference processes of the deep neural network material model. FEM: Finite Element Method.

where $J = \sqrt{\det \mathbf{C}}$ is the volume change, and the isochoric strain invariants, \hat{I}_1 , \hat{I}_2 , \hat{I}_{4v} and \hat{I}_{4w} are defined as

$$\begin{aligned}\hat{I}_1 &= \hat{\mathbf{C}} : \mathbf{I} = \text{tr}(\hat{\mathbf{C}}), \\ \hat{I}_2 &= \frac{1}{2}[\hat{I}_1^2 - \text{tr}(\hat{\mathbf{C}}^2)], \\ \hat{I}_{4v} &= \hat{\mathbf{C}} : \mathbf{v}_0 \otimes \mathbf{v}_0 = \hat{\mathbf{C}} : \mathbf{V}_0, \\ \hat{I}_{4w} &= \hat{\mathbf{C}} : \mathbf{w}_0 \otimes \mathbf{w}_0 = \hat{\mathbf{C}} : \mathbf{W}_0.\end{aligned}\quad (2)$$

The isochoric right Cauchy Green deformation, $\hat{\mathbf{C}}$, can be defined in terms of the isochoric part of the deformation gradient,

$$\hat{\mathbf{F}} = J^{-1/3} \mathbf{F}, \quad (3)$$

$$\hat{\mathbf{C}} = \hat{\mathbf{F}}^\top \hat{\mathbf{F}} = J^{-2/3} \mathbf{C}. \quad (4)$$

The second Piola-Kirchhoff stress tensor, \mathbf{S} , follows from the Doyle-Erickson formula by differentiating the strain energy Ψ with respect to \mathbf{C} and following

the procedure outlined by Coleman and Noll [28, 29], arriving at

$$\mathbf{S} = 2 \frac{\partial \Psi}{\partial \mathbf{C}} = \mathbf{S}_{\text{iso}} + \mathbf{S}_{\text{vol}}, \quad (5)$$

$$\mathbf{S}_{\text{iso}} = \hat{\mathbf{S}} : \frac{\partial \hat{\mathbf{C}}}{\partial \mathbf{C}} = J^{-2/3} \hat{\mathbf{S}} : \mathbb{P}_1, \quad \mathbf{S}_{\text{vol}} = 2p \frac{\partial J}{\partial \mathbf{C}} = J p \mathbf{C}^{-1}. \quad (6)$$

The following definition of the pressure has been introduced $p = d\Psi_{\text{vol}}/dJ$. Additionally, the fictitious second Piola-Kirchhoff stress tensor, $\hat{\mathbf{S}}$, is the result from differentiating the isochoric part of the strain energy with respect to the isochoric invariants, i.e. $\hat{\Psi}_1 = \partial\Psi_{\text{iso}}/\partial\hat{I}_1$, $\hat{\Psi}_2 = \partial\Psi_{\text{iso}}/\partial\hat{I}_2$, $\hat{\Psi}_{4v} = \partial\Psi_{\text{iso}}/\partial\hat{I}_{4v}$ and $\hat{\Psi}_{4w} = \partial\Psi_{\text{iso}}/\partial\hat{I}_{4w}$. The full expansion of the fictitious stress tensor is

$$\hat{\mathbf{S}} = 2 \frac{\partial \Psi_{\text{iso}}}{\partial \hat{\mathbf{C}}} = 2[\hat{\Psi}_1 \mathbf{I} + \hat{\Psi}_2 (\hat{I}_1 \mathbf{I} - \hat{\mathbf{C}}) + \hat{\Psi}_{4v} \mathbf{V}_0 + \hat{\Psi}_{4w} \mathbf{W}_0]. \quad (7)$$

The term *fictitious* originates from the fact that derivatives with respect to the full right Cauchy Green deformation tensor needs the projection with the fourth order tensor

$$\mathbb{P}_1 = \mathbb{I} - \frac{1}{3} \mathbf{C}^{-1} \otimes \mathbf{C},$$

which relates derivatives with respect to the isochoric part of the deformation to derivatives with respect to the total deformation. **Note that if the material under consideration is incompressible**, i.e. $J = 1$, then $\hat{\mathbf{C}} = \mathbf{C}$, (5) reduces to

$$\mathbf{S} = 2[\hat{\Psi}_1 \mathbf{I} + \hat{\Psi}_2 (\hat{I}_1 \mathbf{I} - \mathbf{C}) + \hat{\Psi}_{4v} \mathbf{V}_0 + \hat{\Psi}_{4w} \mathbf{W}_0] + p \mathbf{C}^{-1}, \quad (8)$$

and the pressure p becomes an unknown Lagrange multiplier field. In certain cases, p can be solved from boundary conditions. In this study, the nearly incompressible formulation is used in the finite element formulation, while the incompressible formulation is used during neural network training since this constraint can be easily enforce for the plane stress biaxial deformations considered.

The UMAT subroutine requires the computation of the Cauchy stress tensor, σ . Thus, for completeness, we state the standard push forward operation for the stress

$$\sigma = \frac{1}{J} \mathbf{F} \mathbf{S} \mathbf{F}^T. \quad (9)$$

The finite element subroutine also requires the computation of the elasticity tensor, $\mathbb{C}_{\text{abaqus}}$ [30]. For ease of derivation, the material version of the elasticity tensor, \mathbb{C} , is introduced first,

$$\mathbb{C} = 2 \frac{\partial \mathbf{S}}{\partial \mathbf{C}} = \mathbb{C}_{\text{iso}} + \mathbb{C}_{\text{vol}}, \quad (10)$$

$$\mathbb{C}_{\text{iso}} = 2 \frac{\partial \mathbf{S}_{\text{iso}}}{\partial \mathbf{C}}, \quad \mathbb{C}_{\text{vol}} = 2 \frac{\partial \mathbf{S}_{\text{vol}}}{\partial \mathbf{C}}. \quad (11)$$

6 Data-driven Soft Tissue Mechanics

The expressions for the volumetric and isochoric parts of the elasticity tensor, \mathbb{C}_{vol} and \mathbb{C}_{iso} , can be further expanded,

$$\mathbb{C}_{\text{vol}} = J\tilde{p}\mathbf{C}^{-1} \otimes \mathbf{C}^{-1} - 2Jp\mathbf{C}^{-1} \odot \mathbf{C}^{-1}, \quad (12)$$

$$\mathbb{C}_{\text{iso}} = -\frac{2}{3}\mathbf{S}_{\text{iso}} \otimes \mathbf{C}^{-1} + J^{-4/3}\mathbb{P}_1 : \hat{\mathbb{C}} : \mathbb{P}_1^T - \frac{2}{3}\mathbf{C}^{-1} \otimes \mathbf{S}_{\text{iso}} + \frac{2}{3}J^{-2/3}\text{tr}(\hat{\mathbf{S}})\mathbb{P}_2, \quad (13)$$

where the modified pressure term, $\tilde{p} = p + Jdp/dJ$, has been introduced, as well as the special product noted by (\odot) defined as $(\bullet \odot \circ)_{ijkl} = [(\bullet)_{ik}(\circ)_{jl} + (\bullet)_{il}(\circ)_{jk}]/2$, and an additional fourth order projection tensor \mathbb{P}_2 ,

$$\mathbb{P}_2 = \mathbf{C}^{-1} \odot \mathbf{C}^{-1} - \frac{1}{3}\mathbf{C}^{-1} \otimes \mathbf{C}^{-1}.$$

Finally, the fictitious elasticity tensor, $\hat{\mathbb{C}}$ is obtained from differentiating the fictitious stress tensor with respect to the isochoric part of the deformation tensor,

$$\hat{\mathbb{C}} = 2\frac{\partial \hat{\mathbf{S}}}{\partial \hat{\mathbb{C}}}.$$

The full expansion of $\hat{\mathbb{C}}$ is available in the Supplement. The only remark needed in the main text is that the tensor $\hat{\mathbb{C}}$ requires the second derivatives of the strain energy function with respect to the isochoric invariants: $\hat{\Psi}_{11} = \partial^2 \hat{\Psi} / \partial \hat{I}_1^2$, $\hat{\Psi}_{12} = \partial^2 \hat{\Psi} / \partial \hat{I}_2 \partial \hat{I}_1$, $\hat{\Psi}_{14v} = \partial^2 \hat{\Psi} / \partial \hat{I}_{4v} \partial \hat{I}_1$, etc. This point will become important in the design in the neural network later on.

As stated above, the elasticity tensor needed in the UMAT subroutine is associated with the deformed configuration. The push-forward operation for the elasticity tensor yields

$$\mathbb{c} = \frac{1}{J}(\mathbf{F} \overline{\otimes} \mathbf{F}) : \mathbb{C} : (\mathbf{F} \overline{\otimes} \mathbf{F})^T \quad (14)$$

where we have introduced the modified dyadic product defined as $(\bullet \overline{\otimes} \circ)_{ijkl} = (\bullet)_{ik}(\circ)_{jl}$. The tensor \mathbb{c} is related to the Truesdell stress rate; however, Abaqus increments employ the Jaumann stress rate. Therefore, the consistent tangent for Abaqus is not Eq. (14) but rather

$$\mathbb{c}_{\text{abaqus}} = \mathbb{c} + \frac{1}{2}(\sigma \overline{\otimes} \mathbf{I} + \sigma \underline{\otimes} \mathbf{I} + \mathbf{I} \overline{\otimes} \sigma + \mathbf{I} \underline{\otimes} \sigma). \quad (15)$$

with the modified dyadic product $(\bullet \underline{\otimes} \circ)_{ijkl} = (\bullet)_{il}(\circ)_{jk}$.

As remarked before, incompressibility is imposed exactly during the training of the neural network, with p determined from boundary conditions. However, in the UMAT we use a volumetric strain energy which leads to the following expression for p ,

$$p = K(J - 1), \quad (16)$$

with K the bulk modulus. In this study we set $K = 1$ MPa.

10
11
12
13
14
15
16
17
18
19
20
21

Neural network structure and training

We use a fully connected DNN to learn the mechanical behavior of skin. The neural network takes four inputs, the isochoric strain invariants in (2), and produces five outputs, the strain energy, $\Psi_{\text{iso}}^{\text{P}}$, and its derivatives with respect to the invariants, $\hat{\Psi}_1^{\text{P}}$, $\hat{\Psi}_2^{\text{P}}$, $\hat{\Psi}_{4v}^{\text{P}}$ and $\hat{\Psi}_{4w}^{\text{P}}$. Note that the notation $(\bullet)^{\text{P}}$ is used to denote the predicted values of the DNN. The network architecture is summarized in Table 1.

Training data for the DNN is in the form of stretch and stress data, as well as the values of strain energy obtained by the integration of the stretch-stress curves. Therefore, the first component of the loss function is simply the comparison of the predicted strain energy, $\Psi_{\text{iso}}^{\text{P}}$, against the observed $\Psi_{\text{iso}}^{\text{d}}$, where $(\bullet)^{\text{d}}$ is used to refer to data. The first component of the loss function is

$$\mathcal{L}_1 = \frac{1}{N} \sum_{n=1}^N \left[\left((\Psi_{\text{iso}}^{\text{P}})^{(n)} - (\Psi_{\text{iso}}^{\text{d}})^{(n)} \right)^2 + \sum_{i=1,2,4v,4w} \left((\hat{\Psi}_i^{\text{bp}})^{(n)} - (\hat{\Psi}_i^{\text{P}})^{(n)} \right)^2 \right], \quad (17)$$

where $(\bullet)^{(n)}$ denotes the n^{th} training point, out of a total of N training points. The second term in eq. (17) is a regularization term that is added to the loss function to enforce that the predicted derivatives are consistent with the strain energy. In other words, the derivatives that are a direct output of the neural network, $\hat{\Psi}_i^{\text{P}}$, should coincide with the derivatives of $\Psi_{\text{iso}}^{\text{P}}$ calculated using back-propagation, denoted as $\hat{\Psi}_i^{\text{bp}}$ in eq. (17), with $i = 1, 2, 4v, 4w$, and where $(\bullet)^{\text{bp}}$ stands for "back-propagation".

The second component of the loss results from comparing the stress computed with the neural network outputs against the observed stress σ^{d} . The stress, defined in Eqs. (8) and (9), is computed based on the direct strain energy derivatives output by the neural network, Ψ_i^{P} , to produce σ^{P} . The loss for the stress data can then be simply stated as

$$\mathcal{L}_2 = \frac{1}{N} \sum_{n=1}^N \left\| (\sigma^{\text{P}})^{(n)} - (\sigma^{\text{d}})^{(n)} \right\|_F \quad (18)$$

where $(\bullet)_F$ denotes the Frobenius norm.

To guarantee existence of a solution to the boundary value problem, a suitable constraint on the strain energy is that of polyconvexity with respect to the deformation gradient, \mathbf{F} [21, 31, 32]. An alternative approach is to enforce convexity of the strain energy with respect to \mathbf{C} [33]. Convexity with respect to \mathbf{C} can lead to the existence of global minima in boundary value problems under certain conditions [34]. This convexity condition has been employed in constitutive modeling of biological tissues [35] and in numerous studies on data-driven models of hyperelastic materials [19, 33]. For instance, the popular constitutive model by Holzapfel, Gasser and Ogden to capture the mechanical behavior of collagenous tissues was developed to fulfill this condition [5]. However, convexity with respect to \mathbf{C} , and polyconvexity with respect to \mathbf{F}

63
64
65

8 Data-driven Soft Tissue Mechanics

are not equivalent. Polyconvexity of the strain energy function with respect to \mathbf{F} , together with some growth conditions on the strain energy, guarantees the existence of global minimizers to the total potential energy functional [31].

In the current study we enforce convexity of the strain energy function with respect to the deformation invariants of \mathbf{C} with a global minimum at or below the point $I_1 = 3, I_2 = 3, I_{4v} = 1$ and $I_{4w} = 1$. The invariants are already convex functions of \mathbf{F} or $\text{cof}\mathbf{F}$ [21, 36]. Thus, the non-decreasing convex function of the invariants, for $I_1 \geq 3, I_2 \geq 3, I_{4v} \geq 1$ and $I_{4w} \geq 1$, together with a suitable volumetric energy convex in J results in polyconvexity of the strain energy with respect to \mathbf{F} .

For a function to be convex with respect to its arguments, it's Hessian matrix, \mathbf{H} , must be positive semi-definite [37]. The Hessian matrix of the strain energy as a function of the invariants is

$$\mathbf{H} = \begin{pmatrix} \Psi_{11}^{\text{bp}} & \Psi_{12}^{\text{bp}} & \Psi_{14v}^{\text{bp}} & \Psi_{14w}^{\text{bp}} \\ \Psi_{21}^{\text{bp}} & \Psi_{22}^{\text{bp}} & \Psi_{24v}^{\text{bp}} & \Psi_{24w}^{\text{bp}} \\ \Psi_{4v1}^{\text{bp}} & \Psi_{4v2}^{\text{bp}} & \Psi_{4v4v}^{\text{bp}} & \Psi_{4v4w}^{\text{bp}} \\ \Psi_{4w1}^{\text{bp}} & \Psi_{4w2}^{\text{bp}} & \Psi_{4w4v}^{\text{bp}} & \Psi_{4w4w}^{\text{bp}} \end{pmatrix}. \quad (19)$$

The notation Ψ_{ij}^{bp} indicates the second derivative of the strain energy computed with the neural network by differentiating the outputs $\hat{\Psi}_i^{\text{p}}$ with respect to the j^{th} input using back-propagation. We impose positive-definiteness of the Hessian matrix using the principal minor test [38]. For a matrix to be positive definite, it has to be symmetric and all its leading principal minors, Δ_k , must be positive. This condition is imposed in terms of an additional loss term,

$$\mathcal{L}_3 = \frac{1}{N} \sum_{n=1}^N \left\| (\mathbf{H})^{(n)} - (\mathbf{H}^{\top})^{(n)} \right\|_F + \frac{1}{N} \sum_{n=1}^N \sum_{k=1}^4 \max \left((-\Delta_k^{(n)}, 0) \right). \quad (20)$$

Note that non-negative derivatives are obtained by passing the outputs $\hat{\Psi}_i^{\text{p}}$ through the quadratic function $y_i = x_i^2$ as indicated in Table 1. The choice of Linear or Quadratic activation functions in the last layer is selected in the examples below depending on whether convexity is imposed or not. The non-negative outputs do not directly enforce convexity. Rather, given that the invariants considered are convex functions of \mathbf{F} or $\text{cof}\mathbf{F}$, convex non-decreasing functions of these invariants are needed for polyconvexity. The quadratic activation functions for the $\hat{\Psi}_i^{\text{p}}$ outputs in the last layer enforce the non-decreasing condition for the strain energy by restricting the derivatives to be non-negative.

The total loss is a weighted sum of the terms discussed so far,

$$\mathcal{L} = a_1 \mathcal{L}_1 + a_2 \mathcal{L}_2 + a_3 \mathcal{L}_3. \quad (21)$$

If training data from sources with different fidelities are used, the total loss of the multi-fidelity (mf) dataset is given as a weighted sum of the losses of the high fidelity (hf) and low fidelity (lf) datasets,

$$\mathcal{L}_{\text{mf}} = \mathcal{L}_{\text{lf}} + a_{\text{hf}} \mathcal{L}_{\text{hf}}. \quad (22)$$

The training of the DNN was performed using the Adam optimization algorithm [39]. The initial learning rate was set to 4.0e-5. The exponential decay rates for first and second moment estimates, β_1 and β_2 , were set to 0.9 and 0.99 respectively. The DNN was trained in 100000 epochs without the use of batching. The training was implemented using Keras [40] with a Tensorflow [41] back-end on a workstation with the following specifications: Intel Xeon E5-1630 3.70 GHz CPU, 16 GB DDR4/2400 MHz random access memory, and Nvidia GeForce GTX 1080 GPU. The values of the three weights a_1 , a_2 and a_3 are set to 0.1, 1.0 and 0.008, respectively, after performing a hyperparameter study as reported in the Supplement.

Table 1 Neural network architecture

Layer	Number of nodes	Activation function
Input	4	None
Hidden layer 1	4	Sigmoid
Hidden layer 2	8	Sigmoid
Hidden layer 3	8	Sigmoid
Output	5	Linear/Quadratic

Synthetic data generation

In the majority of biomedical applications it is difficult to obtain sufficient high fidelity data to train a neural network. The number of measurements might be limited, or the data points may be constrained to a narrow region of the input space. It is then beneficial to make use of low fidelity data if available.

In this study, high fidelity data are in the form of biaxial stress-stretch measurements. However, only two or three curves within the four-dimensional input space defined by the invariants is explored. Therefore, we generate synthetic data using the Gasser-Ogden-Holzapfel (GOH) [6] material model. The GOH model proposes an isochoric strain energy of the form

$$\hat{\Psi}(\mathbf{C}, \mathbf{a}_0) = \hat{\Psi}_{\text{iso}}(\mathbf{C}) + \hat{\Psi}_{\text{aniso}}(\mathbf{C}, \mathbf{v}_0) \quad (23)$$

where $\mathbf{a}_0 = (\sin \theta, \cos \theta, 0)$ is vector denoting the mean fiber direction and parameterized by the angle θ . The functional forms for the GOH strain energy are

$$\hat{\Psi}_{\text{iso}}(\mathbf{C}) = \mu(\hat{I}_1 - 3), \quad (24)$$

10 *Data-driven Soft Tissue Mechanics*

14 $\hat{\Psi}_{\text{aniso}}(\mathbf{C}, \mathbf{a}_0) = \frac{k_1}{4k_2} [\exp(k_2 E^2) - 1] , \quad (25)$
 15

16 with the generalized fiber strain
 17

18 $E = [\kappa \hat{I}_1 + (1 - 3\kappa) \hat{I}_{4v} - 1] . \quad (26)$
 19

21 The volumetric term that is the same as the one we used to penalize volume
 22 changes in our formulation [5, 6, 42],
 23

24 $\Psi_{\text{vol}} = \frac{K}{2}(J - 1)^2 . \quad (27)$
 25

27 The derivation of the stress tensor for the GOH strain energy is not
 28 repeated here, the interested reader is referred to [6, 25].
 29

30 Synthetic data with the GOH model is generated by fitting the free parameters
 31 to the experimental data using the BFGS optimization algorithm in SciPy
 32 [43].
 33

Finite element method implementation

35 We implemented a general neural network material model in a user material
 36 subroutine (UMAT) in the nonlinear finite element package Abaqus. The sub-
 37 routine was written with minimal assumptions to allow for maximal flexibility.
 38 The neural network structure, weights and biases, activation functions, etc.
 39 are all imported into the subroutine through the input file.
 40

41 The subroutine performs the following tasks:
 42

- 43 1. Read in the architecture, weights and biases, activation function types, etc.,
 as a set of material properties.
- 44 2. Pre-process the deformation gradient to obtain the isochoric invariants in
 Eqs. (2 - 4).
- 45 3. Perform the forward propagation of the neural network to obtain the
 predicted strain energy Ψ^p and its first derivatives Ψ_i^p .
- 46 4. Calculate stress using Eqs. (5 - 7, 9).
- 47 5. Calculate the second derivatives Ψ_{ij}^{bp} with back-propagation.
- 48 6. Compute the consistent tangent c_{abaqus} using Eq. (15).
 49

52 For the forward propagation, let $\mathbf{y}_{i-1} \in \mathbb{R}^m$ be the output of layer $i - 1$ of
 53 the neural network with m nodes. Then the output of layer i is given as
 54

55 $\mathbf{y}_i = g_i(\mathbf{W}_i^T \mathbf{y}_{i-1} + \mathbf{B}_i), \quad \mathbf{y}_i \in \mathbb{R}^n \quad (28)$
 56

57 where $g_i : \mathbb{R} \rightarrow \mathbb{R}$ is the element-wise activation function, $\mathbf{W}_i \in \mathbb{R}^{n \times m}$ is the
 58 weights matrix, and $\mathbf{B}_i \in \mathbb{R}^n$ is the biases vector of the i^{th} layer of the network.
 59 We use the sigmoid activation function in the hidden layers,
 60

61 $g(y) = \frac{1}{1 + e^{-y}}, \quad g'(y) \equiv \frac{dg(y)}{dy} = g(y)(1 - g(y)) . \quad (29)$
 62

10
11
12
13

17 For the derivatives, let $\mathbf{J}_{i-1} \in m \times m_0$ be the matrix containing derivatives of
 18 the nodes of layer $i - 1$ with respect to the inputs of the neural network. Then
 19

$$17 \quad \mathbf{J}_i = \text{diag}(g'(\mathbf{y}_i)) \mathbf{W}_i^T \mathbf{J}_{i-1}, \quad \mathbf{J}_i \in \mathbb{R}^{n \times m_0} \quad (30)$$

20 where m_0 is the number of inputs to the neural network and $\text{diag}(\bullet)$ denotes
 21 a diagonal matrix.
 22

23 Biaxial stress-stretch experiments on porcine and murine 24 skin

25 We use experimental data from biaxial stress-stretch experiments performed on
 26 murine [24] and porcine skin for the training and validation of neural networks.
 27 The data are collected in up to 5 different experimental protocols which are
 28 defined in Table 2.

29 **Table 2** Experimental loading protocols.

30 Loading	λ_x	λ_y	σ_z
31 Off-x	$\sqrt{\lambda}$	λ	0
32 Off-y	λ	$\sqrt{\lambda}$	0
33 Equibiaxial	λ	λ	0
34 Strip-x	λ	1	0
35 Strip-y	1	λ	0

40 Training data

41 High fidelity training data used in this study consists of 13 sets of experimental
 42 data obtained from 2 pigs and 11 mice. The first porcine dataset consists of
 43 122 data points in the off-x and off-y loading protocols, the second porcine
 44 dataset consists of 402 data points which encompasses all 5 loading protocols
 45 in Table 2. The murine dataset consists of 549 points in the off-x, off-y and
 46 equibiaxial protocols.
 47

48 Low fidelity data was generated using the GOH material model. For each
 49 of the 3 high fidelity datasets, first the free parameters of the GOH model were
 50 fitted to the data. Then the model was used to generate 225 synthetic data
 51 points for each of the porcine datasets and 165 points for the murine dataset.
 52

53 During training of the neural networks, the contribution of the high fidelity
 54 data are weighted higher than the low fidelity data. This guides the neural
 55 network to adhere to the experimental data more closely while approximating
 56 the low fidelity data in regions with no high fidelity data. In this study the
 57 ratio between the weights was set to 50 : 1.
 58

59
60
61
62
63
64
65

9
10
11 12 *Data-driven Soft Tissue Mechanics*
12
13
14

3 Results

Performance of the neural network against synthetic data

To test the DNN material model, we first train the network using synthetic data only. We generate eleven *curves* in the λ_x, λ_y space by first holding $\lambda_x = 1$ while λ_y is increased gradually to $\lambda_y^{(i)}$, with $i = 1, \dots, 11$. The values for the y -stretch are $\lambda_y^{(i)} \in [1, 1.025, 1.05, \dots, 1.25]$. After reaching the corresponding $\lambda_y^{(i)}$ value, λ_y is held constant while λ_x is gradually increased (Figure 2(a)). These loading curves are representative of the the type of test that can be performed experimentally. On the other hand, the DNN takes as inputs the isochoric strain invariants. The invariant space is 4-dimensional, but we plot a 3-dimensional projection in Figure 2(b). We use the GOH material model to generate synthetic stress data points and train the neural network. Various components of the loss are plotted in Figure 2(c). The predictions of the trained network are plotted against the training data in Figure 2(d)-(f). These results indicate that the DNN is able to recreate almost perfectly the expert constitutive models within the training region.

We also test if the DNN performs well outside the training region. We generate three validation datasets. The first validation dataset is built by randomly sampling $\lambda_x \in [1, 1.25]$, $\lambda_y \in [1, 1.25]$ to construct a diagonal deformation gradient of biaxial deformations not seen during training. Then, to test predictions under shear, which are not directly part of the training data, we construct a data set of deformation gradients of the form

$$\mathbf{F} = \begin{pmatrix} \lambda_x & \gamma_{xy} & 0 \\ \gamma_{xy} & \lambda_y & 0 \\ 0 & 0 & \frac{1}{\lambda_x \lambda_y - \gamma_{xy}^2} \end{pmatrix}. \quad (31)$$

The validation dataset is generated from randomly sampling $\lambda_x \in [1, 1.25]$, $\lambda_y \in [1, 1.25]$, $\gamma_{xy} \in [0, 0.3]$. Lastly, we are interested in the potential of the neural network to extrapolate under biaxial deformations. An additional validation set is constructed by sampling outside the training region: $\lambda_x \in [1, 1.25]$ but $\lambda_y \in [1.25, 1.35]$; $\lambda_y \in [1, 1.25]$ but $\lambda_x \in [1.25, 1.35]$; and $\lambda_x \in [1.25, 1.35]$ and $\lambda_y \in [1.25, 1.35]$. The errors for the validation datasets are shown in Figure 3.

The stress $\hat{\sigma}^P$ and $\hat{\sigma}^d$ are normalized such that each entry of these tensors is obtained by subtracting the mean and dividing by the standard deviation of the stress values over the validation dataset, e.g. $\hat{\sigma}_{ij}^P = (\sigma_{ij}^P - \sigma_{ij}^{d,\text{avg}})/\Sigma_{ij}^d$, with $\sigma_{ij}^{d,\text{avg}}$ the mean of that stress component over the validation data, and Σ_{ij}^d the corresponding standard deviation. If the data were normal, then the normalized quantities would be almost entirely in the range $[-3, 3]$. Even though the data are not normal, this is a useful scaling of the stress. Relative errors are

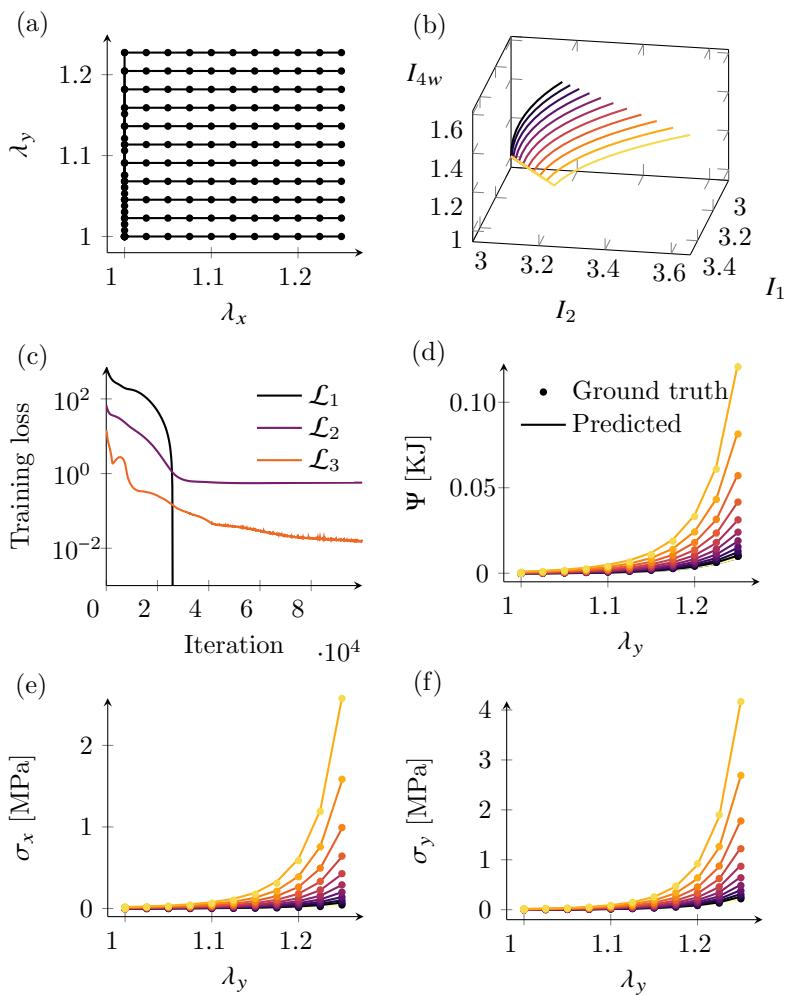


Fig. 2 Synthetic data generated to train the neural network and performance of the neural network compared to the training data. (a) Training data was generated by creating curves in the λ_x, λ_y space. (b) Corresponding training data in the invariant space, which is the actual input space for the neural network. The invariant space is four-dimensional but only a three-dimensional projection is shown. The colors of the curves indicate the value of the fourth invariant. (c) Loss during training. (d) Predicted and ground truth strain energy values. Predicted and ground truth planar stress values in the (e) x - and (f) y -directions. The colors of the curves in (d), (e) and (f) indicate the value of λ_x .

high in the low strain region for which the stress is negligible (orders of magnitude lower than in the high stress regions), and absolute errors are higher in regions of high stress even if the relative error is small. The normalized stress better captures the performance of the DNN over the input space. It can be seen that the DNN performs well within the training region but worse toward the boundary of the training region.

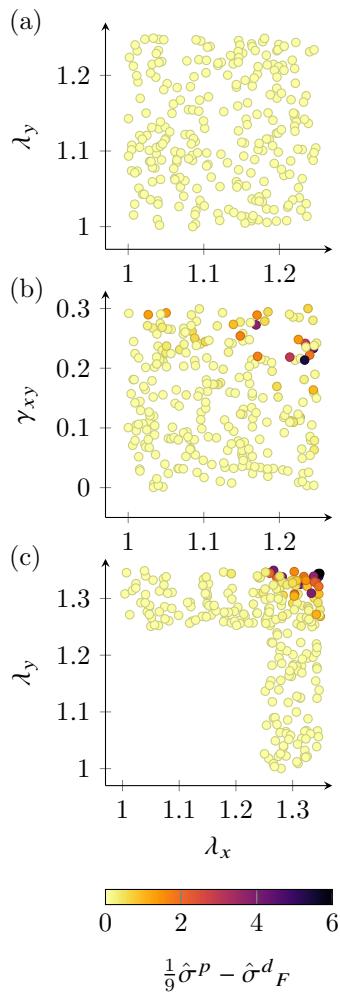


Fig. 3 Validation of the neural network trained on synthetic data. Performance of the neural network on points randomly sampled from: (a) $\lambda_x \in [1, 1.25]$ and $\lambda_y \in [1, 1.25]$, (b) $\lambda_x \in [1, 1.25]$, $\lambda_y \in [1, 1.25]$, $\gamma_{xy} \in [0, 0.3]$ and (c) $\lambda_x \in [1, 1.25]$ but $\lambda_y \in [1.25, 1.35]$; $\lambda_y \in [1, 1.25]$ but $\lambda_x \in [1.25, 1.35]$; and $\lambda_x \in [1.25, 1.35]$ but $\lambda_y \in [1.25, 1.35]$. The colorbar indicates the error of each point as defined at the bottom of the figure.

Performance against experimental data: multi-fidelity data and convexity constraints

Next, we start training the DNN using experimental data. We want to test the effect of using the experimental data alone (sparse high fidelity data), or combining these data with the low fidelity approximation of the GOH model fit (multi-fidelity data). Concurrently, we want to test if the polyconvexity constraint is required to regularize the fits of the DNN. In Figs. 4 and 5, we show the results for murine skin data and porcine skin data respectively

together with error in the predictions which is defined as the average Frobenius norm of the error in stress, $\text{mean}(\|\sigma^P - \sigma\|_F)$.

The first row of Figure 4 corresponds to a neural network that is trained using sparse high-fidelity data where the convexity constraints are not imposed. Figure 4(a) shows the DNN ability to fit the off-x and off-y data, achieving average errors of 4.56 kPa and 5.25 kPa, respectively. A biaxial test, not used in training, is used to test the predictive capability of the network (Figure 4(b)). The average error in the validation is 7.33 kPa. Because no convexity constraint is used we can see that it is not satisfied (Figure 4(c)). Keeping only the high-fidelity data but imposing convexity changes the performance. The training and validation loss are poorer (Figure 4(e)-(g)), but the function is convex over the input space (Figure 4(h)).

The third and fourth rows of Figure 4 show the results of DNNs trained with multi-fidelity data. It is notable that even though the network of the third row is trained without any convexity constraints, the fact that it is trained on the GOH synthetic data (which is an inherently convex model) helps it achieve better convexity (Figure 4(l)). The average error in the validation set for the multi-fidelity case without convexity constraint is 11.75 kPa (Figure 4(k)), which is worse than the sparse high fidelity case without convexity requirements (Figure 4(c)). The results for the neural network trained with multi-fidelity data and with the convexity constraint are shown in last row of Figure 4. The training errors are 8.21 kPa and 7.27 kPa (Figure 4(m) and (n)), and the validation error is 11.47 kPa (Figure 4(o)). Since the convexity is imposed, the loss in the convexity is close to zero over the entire input space (Figure 4(p)). In summary, polyconvexity is a useful framework to guarantee existence of minimizers for problems in elasticity, but it can slightly increase the error of the neural network over the training set. This can be reflective of the uncertainties in the experimental data collection, or limitations of the hyperelastic framework. Additional 10 murine datasets are shown in the Supplemental Material, showing that the DNN can easily fit a wide variety of skin samples.

In Figure 5 we further study the effects of augmenting the training data and how the neural network differs from relying solely on the expert model. For this we focus on porcine skin. We train two DNNs, one of them is trained with the experimental data only (first and third columns of Figure 5), whereas the other is trained on the augmented data (second and fourth columns of Figure 5). In Figure 5(c) it can be seen that trained only on experimental data, the neural network achieves a low average error of 5.13 kPa, compared to the GOH fit which is 22.54 kPa. Thus, the neural network outperforms the GOH model. This is not surprising since the only task of the neural network is to interpolate the experimental data and satisfy convexity. The contour plots in Figure 5(g)-(l) shows the difference between the neural network as compared with GOH material model throughout the input space. It can be seen that the two models differ toward the boundary of the deformation space considered. Surprisingly, the models agree with each other in large portions of the input

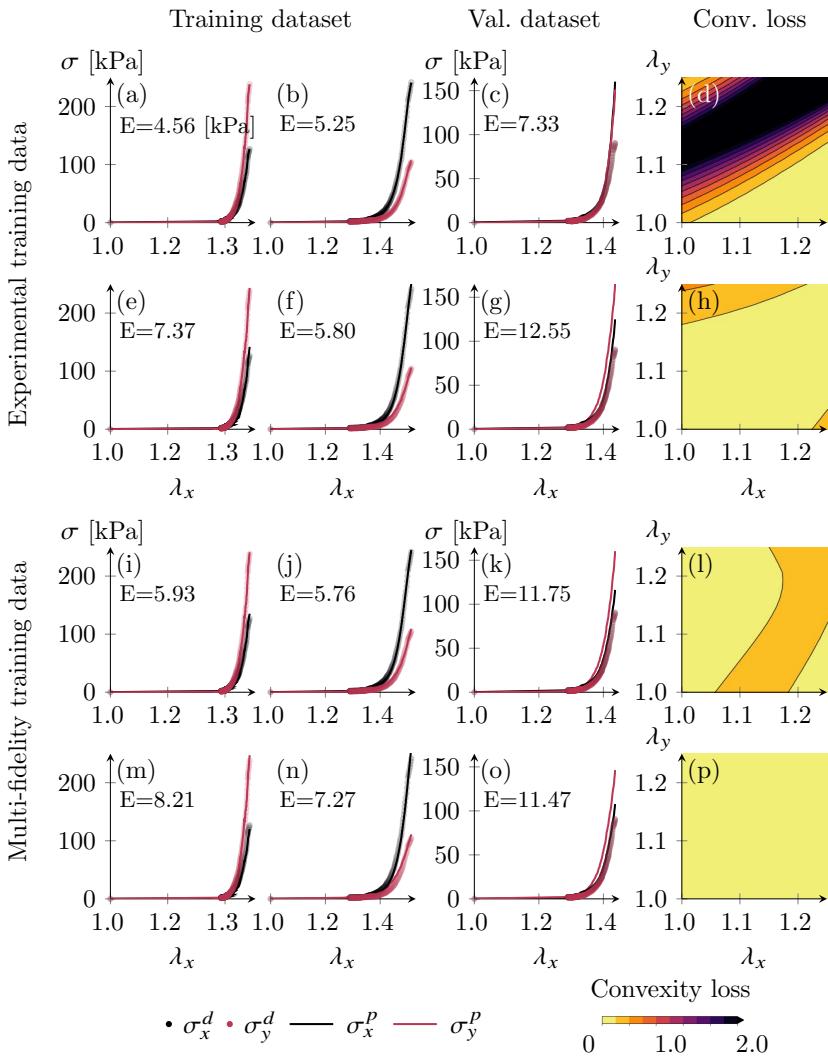
16 *Data-driven Soft Tissue Mechanics*

Fig. 4 Performance of the DNN on the murine skin data and average prediction error, E. The plots in each row show the predicted stress vs actual stress on the training (Off-x and Off-y) and validation (Equibiaxial), while the last column shows the convexity loss throughout the input space. Each row corresponds to a separate DNN. Predictions of DNN trained with (top to bottom): single-fidelity data and no convexity constraints, single-fidelity data and convexity constraints, multi-fidelity data and no convexity constraints, multi-fidelity data and convexity constraints.

space even when the DNN is trained independently of the GOH model (Figure 5(g)-(k)). Based on the validation examples with the synthetic dataset, we know that the DNN does not extrapolate well outside of the training region. On the other hand, the GOH model has been developed and trained against thousands of tissue biaxial data. It is reasonable to expect that the GOH model, even though it cannot fit any particular dataset as well as the DNN, it

can be trusted to guide the neural network away from the training region. We show that training the neural network on the augmented data, the loss is on average 12.81 kPa against the experimental data (Figure 5(c) and (e)), which is higher than the single-fidelity DNN but still lower with respect to using the GOH model alone. However, as looking at the contours in Figure 5(h), (j) and (l) we see that the neural network now follows the GOH model even more closely on the entire input space.

Therefore, the DNN trained with augmented data are at the very least the best version of the GOH model. It performs better than the GOH material model around the high fidelity data points while approximating the GOH model elsewhere.

The last test of the DNN material model is also done with porcine experimental data. In this case we have five different biaxial experiments (see Table 2). We are interested in determining which biaxial tests are the most informative for the DNN material model. Thus, we train the DNN with different combination of experimental data and validate against the rest of the data (Figure 6). We do the same training and testing with the GOH model. In Figure 6(a) and (b) we train the material models with only two datasets, and test against the other three. In the training set, as expected from our previous result, the neural network outperforms the GOH model. In the validation data we see that the neural network performs similarly to the GOH model in the case in which it is trained on Off-x and Off-y data, but is outperformed by the GOH model in the case in which it is trained with Strip-x and Strip-y data. Figure 6(c), (d) show the result of training the models with three of the five biaxial curves, and validating against the remaining two. Again, the DNN obviously outperforms the GOH model in the training set. During validation, when the DNN is trained on strip biaxial data as well as equi-biaxial data it is able to outperform the GOH model in both validation cases (Figure 6(d)). In Figure 6(e) we train against four tests, and validate against the equibiaxial test. The validation and training errors are lower for the neural network compared to the GOH model. The superior performance of the DNN on the last case confirms that data-driven models are a preferable alternative to expert constructed constitutive models when sufficient training data are available.

Finite element method implementation

We show a number of basic finite element simulations to test the capabilities of the DNN material model as a UMAT subroutine for Abaqus. The neural network trained on porcine skin data (with convexity constraint) is defined in the input file. We first consider a rectangular block $5 \times 5 \times 1$ cm³. Boundary conditions, mesh and results for a uniaxial extension simulation with $\lambda_x = 1.2$ are depicted in Figure 7(a). The result is a homogeneous stress distribution of $\sigma_x = 29.9$ kPa, $\sigma_y = \sigma_z = 0$, consistent with the results in Fig. 6, confirming that the UMAT subroutine is working as intended.

A shearing simulation is shown in Figure 7(b). In this analysis the -x surface of the prism is clamped and a displacement boundary condition of $U_x = U_y = 5$

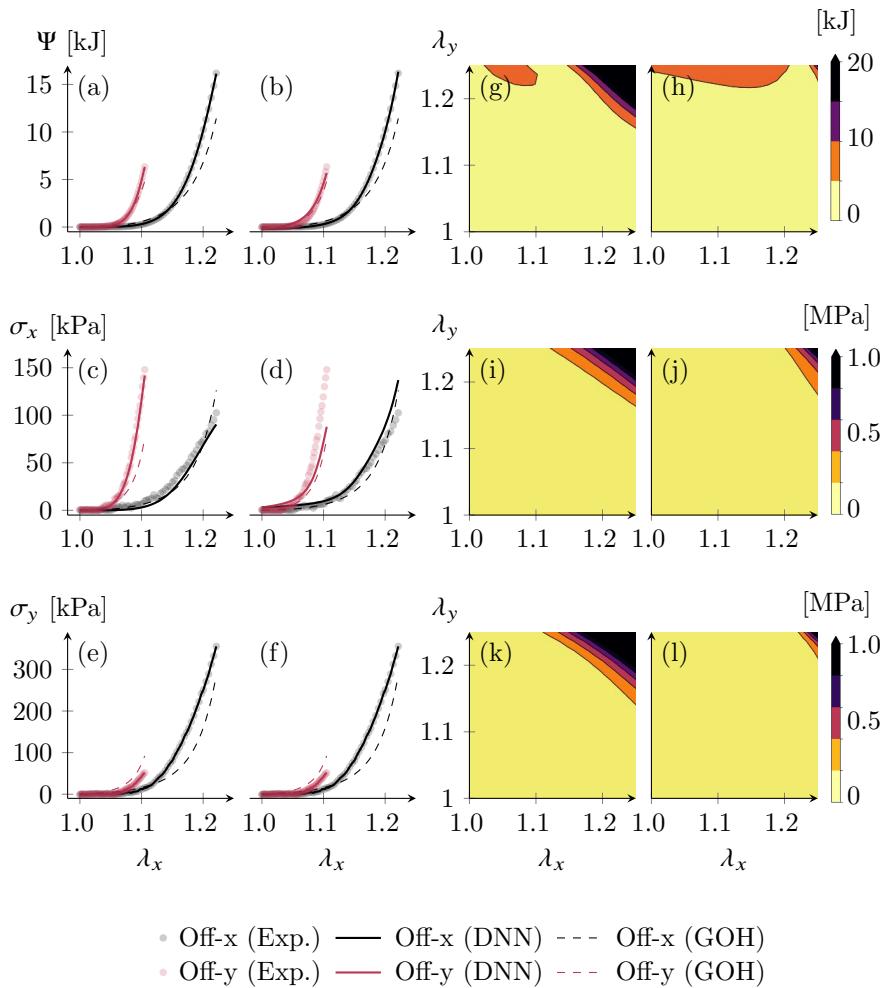


Fig. 5 Performance of DNNs trained on single-fidelity (first and third columns) and multi-fidelity (second and fourth columns) training data. The scatter plots compare predicted strain energy and stress values to experimental data as well as GOH model outputs. The contour plots show the difference between the corresponding outputs of the GOH model and the DNN.

is applied on the right surface. The contours of the resulting stress components, σ_x and σ_y are shown in Figure 7. The Supplement shows a simulation with the GOH fit. As discussed in the previous section, the neural network model with the augmented data are, in a way, the best extension of the GOH model: it retains some of the expert model features but does not suffer from the constraints of an explicit functional form.

The last simulation in Figure 7 is a torsional loading scenario. In this simulation the $-x$ surface of the rectangular prism is clamped and a rotation boundary condition of $UR_x = 1 \text{ rad}$ is imposed on the $+x$ surface. The resulting

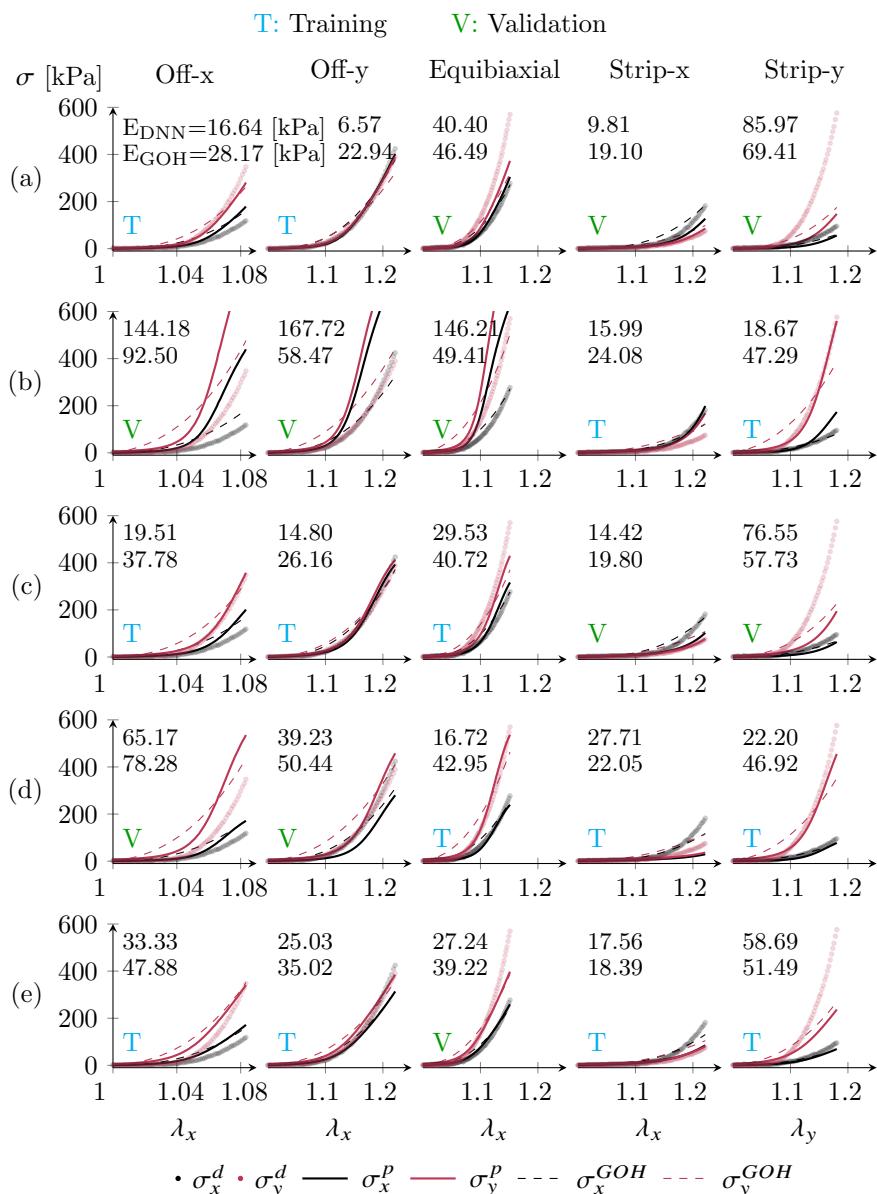


Fig. 6 Comparison of predicted stress vs actual stress vs GOH fits for various training-/validation splits of porcine experimental data. Predictions of a neural network trained on (a) Off-x and Off-y data, (b) Strip-x and Strip-y data, (c) Off-x, Off-y and Equibiaxial data, (d) Equibiaxial, Strip-x and Strip-y data and (e) Off-x, Off-y, Strip-x and Strip-y data.

9
10
11 20 *Data-driven Soft Tissue Mechanics*
12
13

14 stresses are presented in Figure 7(c). This loading scenario is different from
15 the previous two because it involves significant deformations in the out-of-
16 plane direction. The UMAT subroutine executes without any problems. The
17 three simulations in Figure 7 showcase the robustness and versatility of our
18 DNN UMAT.
It should also be noted that the DNN material model usually requires approximately twice as much computational time compared to the built-in GOH model. For example, for the torsion problem, the execution time for GOH is 00:01:42, while for the DNN UMAT it is 00:02:24.

22 Next, we perform a simulation that is much more closely related to skin
23 biomechanics. Tissue expansion is a widely used technique in reconstructive
24 surgery in which a balloon-like device is inserted and inflated subcutaneously
25 to stretch and grow skin [42]. The domain is a $10 \times 10 \times 0.3$ cm³ patch of
26 skin modeled with 3200 brick elements. A rectangular expander of dimensions
27 8×8 cm² underneath the skin mesh is modeled with the fluid cavity feature
28 in Abaqus. The expander is inflated to 20, 40 and 60 cm³ resulting in the
29 principal strain distributions shown in Figure 8. Once again, the simulation
30 converged without issues and the results align with our previous experimental
31 observations of higher deformation at the apex and less toward the periphery
32 of the expander [44]. The simulation in Figure 8 showcases the ability of our
33 neural network model to be used in realistic finite element simulations through
34 our UMAT.
35

36 Note that the simulation in Figure 8 evidences the anisotropy of the model.
37 The fiber directions \mathbf{v} and \mathbf{w} are aligned with the Cartesian basis [1, 0, 0] and
38 [0, 1, 0]. The tissue is stiffer in the \mathbf{v} direction, which is why there is a band
39 of higher stress along that direction in Figure 8. To further showcase the
40 anisotropy in the deformation we also plot the corresponding strains (Figure
41 9).

43
44 4 Discussion
45

46 In this study we propose a deep neural network (DNN) material model to
47 replace conventional constitutive equations for nonlinear materials, in particu-
48 lar soft collagenous tissues. The neural network takes isochoric strain invariants
49 as inputs and produces the isochoric strain energy and its derivatives as
50 outputs. With this design, objectivity is satisfied *a priori*. Other efforts in
51 data-driven modeling of materials and structures rely on training directly on
52 the stress data, which requires additional steps to ensure objectivity [45–47].
For instance, additional loss functions to deal with the violation of objectivity have been proposed [48]. Efforts using invariants or principal stretches as inputs with energy as the output have also been shown by others [19, 49, 50], and by us as well but for isotropic materials [18]. Intermediate approaches that map deformation invariants to principal stresses have also been sought, but they are limited to isotropic materials and even in that case require regularization schemes [51]. The key ideas introduced in this paper are the consideration
53 of anisotropy, training with multi-fidelity data -including experimental data-,
54 and the use of a physics-informed loss function to ensure objectivity.

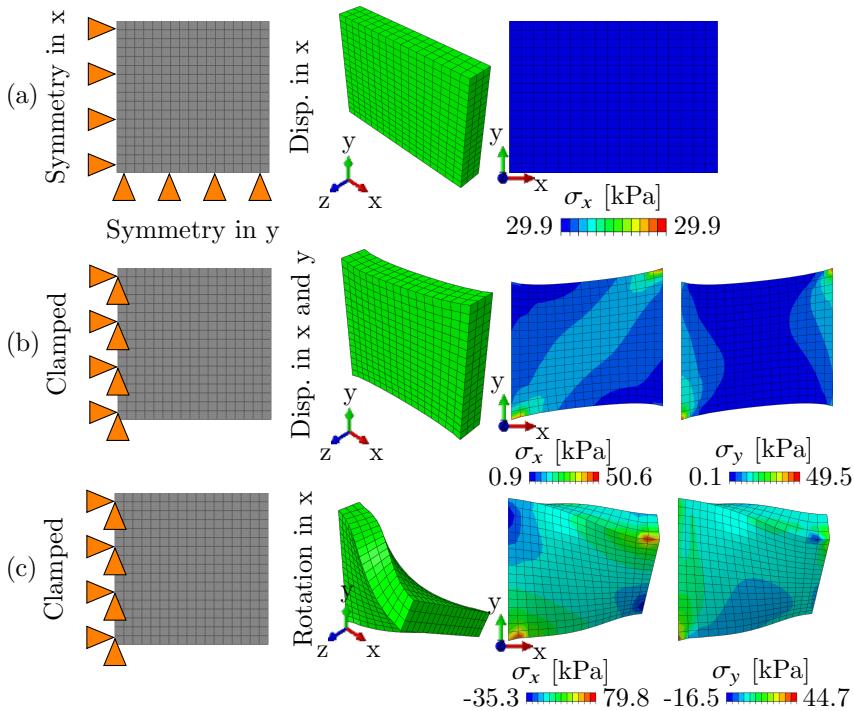


Fig. 7 Finite element method simulations using the DNN material model in UMAT. Boundary conditions, deformed geometry and contours of σ_x under uniaxial loading (a), Boundary conditions, deformed geometry and contours of σ_x and σ_y under shear loading (b), and, Boundary conditions, deformed geometry and contours of σ_x and σ_y under torsional loading (c).

polyconvexity constraints, and design of the DNN architecture to compute not only the stress but the consistent tangent needed in finite element simulations.

Training data for the neural network can consist of both *expensive* (or hard to get) high fidelity data such as results of laboratory experiments, or a combination of high fidelity data supplemented by synthetic data from expert models. A control case shown here is to train the neural network on synthetic data alone. The performance of the neural network with the synthetic data shows that the network can interpolate the expert model almost perfectly (Fig. 2). This is not surprising since neural networks are universal approximators [52]. Other work using neural network material models have also shown excellent performance against synthetic data [53]. When enough data are available, recent efforts in data-driven computational mechanics have shown that model-free approaches can be used [47, 54]. However, for applications in biomechanics, the anisotropy and high nonlinearity in the materials necessitates large amounts of data from deformations that can cover the entire input space [55, 56]. This is often out of reach soft tissue characterization. Thus, we

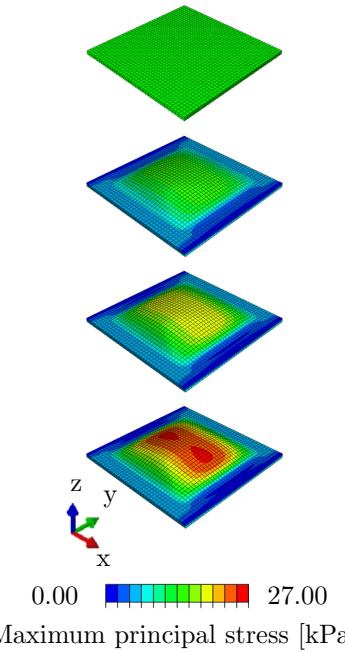
22 *Data-driven Soft Tissue Mechanics*

Fig. 8 Finite element method simulations of tissue expansion using the DNN material model in UMAT. From **top to bottom**: Undeformed geometry, and, contours of maximum principal stress on deformed geometry after the expander is expanded to 20, 40 and 60 cm³, respectively.

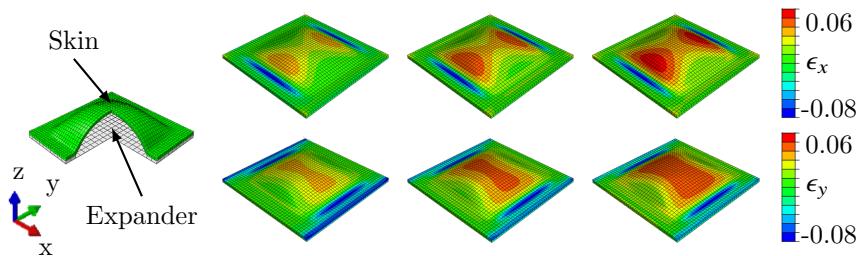


Fig. 9 Finite element method simulations of tissue expansion using the DNN material model in the UMAT. From left to right: model setup, and, contours of strain on deformed geometry after the expander is expanded to 20, 40 and 60 cm³, respectively.

propose a DNN model that captures the experimental data, but does so constrained by a hyperelastic framework and the condition of polyconvexity of the strain energy.

High fidelity data of soft tissue mechanics is sparse in most applications. In our previous work, typically only three protocols have been performed: off-biaxial x, off-biaxial y and equi-biaxial [24]. Two other tests, strip-biaxial tests

in the x- and y-direction are explored here as well. Liu et al. [19] generated datasets by subjecting tissues to seven biaxial tests. Clearly, more coverage of the input space is always better for data-driven approaches. However, this is a challenge, it requires establishment of multiple repeatable protocols and extensive testing of individual specimens which can introduce unforeseen uncertainties. Within the hyperelastic framework, expert models of soft tissue mechanics have been developed over the past few decades and reflect our growing understanding of soft tissues. For example, expert models are often based on microstructure observations [9, 57], satisfy physics constraints [21, 35], and have been carefully designed based on observations of many data [7]. On the other hand, expert models have many limitations, such as non-uniqueness of fit, high sensitivity to parameters, and inability to fit the data due to the inherent constraints of the functional form [8]. By combining the high fidelity data with an expert model as a low fidelity approximation we aim at getting the best of both, keep data-centric models that can capture the experimental data with great accuracy, while maintaining relatively good performance in regions with scarce high fidelity data. Of course, this raises the question of how to balance between the two. Here we set a much higher priority for the experimental data, but future studies should quantify the uncertainty of both the data and the models in regions with little experimental data in order to rigorously weight the high and low fidelity models. With the current weights, we showcase the ability of the DNN material model to capture the mechanical response of skin based on data obtained from 2 pigs and 11 mice, demonstrating the applicability of our approach to realistic datasets.

Imposing polyconvexity through the loss function ensures a stable material model suitable for finite element applications, but comes at the expense of fitting error (see Figure 4). This result points to polyconvexity as a potentially restrictive condition on the data, possibly due to the existence of dissipative phenomena such as viscoelasticity or damage which were not accounted for in the model [2, 58]. Noise can also affect the performance of the machine learning approaches [59]. The Supplement shows that imposing the convexity constraint, the DNN model can capture the synthetic data even in the presence of noise. Other data-driven approaches have considered different convexity constraints. For example, Vlassis et al. [33] check for convexity of the strain energy with respect to the right Cauchy Green deformation tensor C . While convexity with respect to C is widely used [34], it is not equivalent to polyconvexity with respect to F [60]. Polyconvexity of the strain energy, together with some growth conditions on the energy, ensures the existence of minimizers for boundary value problems in elasticity [31]. While it is true that polyconvexity is a sufficient but not a necessary condition, it provides enough flexibility, is compatible with phenomena such as buckling [61], and is desirable for finite element implementation. Data-driven work enforcing polyconvexity have also been explored with different approaches by us and others [53, 62].

A strong motivation behind the development of data-driven constitutive models of soft tissues based on experimental tissue testing data is to use the

9
10
11
12 24 *Data-driven Soft Tissue Mechanics*
13

14 model in predictive finite element simulations to guide device design or treatment
15 planning [63]. Previous work on data driven modeling has fallen short
16 in this regard [19, 53, 64]. The DNN design shown here, including the use of
17 invariants as inputs and prediction of energy and energy derivatives as outputs,
18 allows us to compute not only the stresses but the consistent tangent. Together
19 with the polyconvexity loss, our data-driven framework is uniquely suited for
20 finite element simulations. While it would be possible to predict the energy
21 alone, this introduces noise in the derivatives that need to be regularized as
22 shown in [33]. An alternative framework is to use integrable neural networks
23 [22]. Another method we have explored recently is the use of neural ordinary
24 differential equations to learn the energy derivatives, ignoring the underlying
25 energy function entirely [62]. In [62], the model architecture guarantees that
26 the derivative functions do indeed come from differentiation of an underlying
27 potential even if this potential is not explicitly modeled. The approach fol-
28 lowed here is more akin to multifield formulations in elasticity or enhanced
29 strain methods, for which additional degrees of freedom are added together
30 with suitable constraints [65]. We implemented the DNN model in a UMAT
31 subroutine for Abaqus, a popular finite element package in both academia and
32 industry. The UMAT subroutine code was implemented with maximum flex-
33 ibility in mind. The definition and all parameters of the neural network are
34 provided to the UMAT through the input file. We showcased finite element
35 simulations with the neural network trained on the porcine data, from simple
36 deformations to realistic applications such as tissue expansion.
37

38 Of course, this work is not without limitations. While it is common to
39 model soft tissues within the hyperelastic framework, other physical phenom-
40 ena will be included in future work, namely viscoelasticity, interstitial flow,
41 and damage. Additionally, a Bayesian framework is needed to account for the
42 inherent uncertainty in material behavior of biological materials. Neverthe-
43 less, we anticipate that the general framework introduced here will open up
44 new avenues in data-driven finite element models that balance high-fidelity
45 experimental data with expert knowledge of soft tissue mechanics.
46

47
48 5 Conclusion
49

50 The work presented in this study shows that neural network material mod-
51 els can reliably replace or augment conventional constitutive material models
52 in tissue mechanics analyses. If enough high fidelity data are available, data-
53 driven models can eliminate the burden of choosing a specific functional form
54 and the inherent limitations that come with this choice. However, in most
55 applications, high fidelity data are scarce. Our work demonstrates that a multi-
56 fidelity approach can leverage expert knowledge in the form of synthetic data,
57 while achieving a better fit to the experimental observations. A strong moti-
58 vation to develop accurate material models of soft tissue is to build predictive
59 finite element models. We designed the neural network with this application in
60

mind, and implemented a DNN UMAT subroutine for Abaqus, a widely used finite element package.

Supplementary information. Murine biaxial stress-strain data are available through Manuel K. Rausch's dataverse (together with other mechanical raw data): <https://dataverse.tdl.org/dataverse/STBML> Code associated with this manuscript is available at: https://github.com/abuganza/NN_aniso_UMAT

Acknowledgments. This work was supported by the National Institute of Arthritis and Musculoskeletal and Skin Diseases, National Institutes of Health, United States under award R01AR074525 to Adrian Buganza Tepole and the National Science Foundation through awards 1916663 and 2127925 to Manuel K. Rausch as well as 1916665 to Adrian Buganza Tepole.

Declarations

The authors declare no competing interests.

References

- [1] Lee, T., Turin, S.Y., Stowers, C., Gosain, A.K., Tepole, A.B.: Personalized computational models of tissue-rearrangement in the scalp predict the mechanical stress signature of rotation flaps. *The Cleft Palate-Craniofacial Journal* **58**(4), 438–445 (2021)
- [2] Sherman, V., Tang, Y., Zhao, S., Yang, W., Meyers, M.: Structural characterization and viscoelastic constitutive modeling of skin. *Acta Biomaterialia* **53**, 460–469 (2017)
- [3] Kakaletsis, S., Meador, W.D., Mathur, M., Sugerman, G.P., Jazwiec, T., Malinowski, M., Lejeune, E., Timek, T.A., Rausch, M.K.: Right ventricular myocardial mechanics: Multi-modal deformation, microstructure, modeling, and comparison to the left ventricle. *Acta Biomaterialia* **123**, 154–166 (2021)
- [4] Meador, W.D., Mathur, M., Sugerman, G.P., Jazwiec, T., Malinowski, M., Bersi, M.R., Timek, T.A., Rausch, M.K.: A detailed mechanical and microstructural analysis of ovine tricuspid valve leaflets. *Acta biomaterialia* **102**, 100–113 (2020)
- [5] Holzapfel, G.A.: Nonlinear Solid Mechanics; A Continuum Approach for Engineering. John Wiley & Sons, LTD, ??? (2000)
- [6] Gasser, T.C., Ogden, R.W., Holzapfel, G.A.: Hyperelastic modelling of arterial layers with distributed collagen fibre orientations. *Journal of the royal society interface* **3**(6), 15–35 (2005)

- [17] Lejeune, E., Zhao, B.: Exploring the potential of transfer learning for metamodels of heterogeneous material deformation. *Journal of the Mechanical Behavior of Biomedical Materials* **117**, 104276 (2021). <https://doi.org/10.1016/j.jmbbm.2020.104276>
- [18] Leng, Y., Calve, S., Tepole, A.B.: Predicting the mechanical properties of fibrin using neural networks trained on discrete fiber network data. arXiv preprint arXiv:2101.11712 (2021)
- [19] Liu, M., Liang, L., Sun, W.: A generic physics-informed neural network-based constitutive model for soft biological tissues. *Computer Methods in Applied Mechanics and Engineering* **372** (2020)
- [20] Reimann, D., Chandra, K., Vajragupta, N., Glasmachers, T., Junker, P., Hartmaier, A., *et al.*: Modeling macroscopic material behavior with machine learning algorithms trained by micromechanical simulations. *Frontiers in Materials* **6**, 181 (2019)
- [21] Ehret, A.E., Itskov, M.: A polyconvex hyperelastic model for fiber-reinforced materials in application to soft tissues. *Journal of Materials Science* **42**(21), 8853–8863 (2007)
- [22] Teichert, G.H., Natarajan, A., Van der Ven, A., Garikipati, K.: Machine learning materials physics: Integrable deep neural networks enable scale bridging by learning free energy functions. *Computer Methods in Applied Mechanics and Engineering* **353**, 201–216 (2019)
- [23] Nguyen, L.T.K., Keip, M.-A.: A data-driven approach to nonlinear elasticity. *Computers & Structures* **194**, 97–115 (2018)
- [24] Meador, W.D., Sugerman, G.P., Story, H.M., Steifert, A.W., Bersi, M.R., Tepole, A.B., Rausch, M.K.: The regional-dependent biaxial behavior of young and aged mouse skin: A detailed histomechanical characterization, residual strain analysis, and constitutive model. *Acta Biomaterialia* **101**, 403–413 (2020)
- [25] Lee, T., Turin, S.Y., Gosain, A.K., Bilionis, I., Tepole, A.B.: Propagation of material behavior uncertainty in a nonlinear finite element model of reconstructive surgery. *Biomechanics and Modeling in Mechanobiology* **17**, 1857–1873 (2018)
- [26] Bonfiglio, L., Perdikaris, P., Brizzolara, S.: Multi-fidelity bayesian optimization of swath hull forms. *Journal of Ship Research* **64**(2), 154–170 (2020)
- [27] Smith, M.: ABAQUS/Standard User's Manual, Version 6.9. Dassault Systèmes Simulia Corp, United States (2009)

28 *Data-driven Soft Tissue Mechanics*

- [28] Doyle, T.C., Ericksen, J.L.: Nonlinear elasticity. *Advances in Applied Mechanics*, vol. 4, pp. 53–115. Elsevier (1956). [https://doi.org/10.1016/S0065-2156\(08\)70371-5](https://doi.org/10.1016/S0065-2156(08)70371-5). <https://www.sciencedirect.com/science/article/pii/S0065215608703715>
- [29] Coleman, B.D., Walter, N.: *The Foundations of Mechanics and Thermodynamics*. Springer, ??? (1974)
- [30] Fehervary, H., Maes, L., Vastmans, J., Kloosterman, G., Famaey, N.: How to implement user-defined fiber-reinforced hyperelastic materials in finite element software. *Journal of the Mechanical Behavior of Biomedical Materials* **110**, 103737 (2020). <https://doi.org/10.1016/j.jmbbm.2020.103737>
- [31] Ball, J.M.: Convexity conditions and existence theorems in nonlinear elasticity. *Archive for rational mechanics and Analysis* **63**(4), 337–403 (1976)
- [32] Schröder, J.: Anisotropic polyconvex energies. In: Poly-, Quasi-and Rank-one Convexity in Applied Mechanics, pp. 53–105. Springer, ??? (2010)
- [33] Vlassis, N.N., Ma, R., Sun, W.: Geometric deep learning for computational mechanics part i: anisotropic hyperelasticity. *Computer Methods in Applied Mechanics and Engineering* **371**, 113299 (2020). <https://doi.org/10.1016/j.cma.2020.113299>
- [34] Gao, D.Y., Neff, P., Roventa, I., Thiel, C.: On the convexity of nonlinear elastic energies in the right cauchy-green tensor. *Journal of Elasticity* **127**(2), 303–308 (2017)
- [35] Holzapfel, G.A., Gasser, T.C., Ogden, R.W.: A new constitutive framework for arterial wall mechanics and a comparative study of material models. *Journal of elasticity and the physical science of solids* **61**(1), 1–48 (2000)
- [36] Schröder, J., Neff, P., Balzani, D.: A variational approach for materially stable anisotropic hyperelasticity. *International journal of solids and structures* **42**(15), 4352–4371 (2005)
- [37] Boyd, S., Vandenberghe, L.: *Convex Optimization*. Cambridge University Press, ??? (2004)
- [38] Prussing, J.E.: The principal minor test for semidefiniteness matrices. *Journal of Guidance, Control and Dynamics* **9**(1), 121–122 (1985)
- [39] Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)

- 17 [40] Chollet, F., et al.: Keras: The python deep learning library. *ascl*, 1806
18 (2018)
- 19 [41] Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Cor-
20 rrado, G.S., Davis, A., Dean, J., Devin, M., et al.: Tensorflow: Large-scale
21 machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467* (2016)
- 22 [42] Lee, T., Turin, S.Y., Gosain, A.K., Bilionis, I., Tepole, A.B.: Propagation
23 of material behavior uncertainty in a nonlinear finite element model of
24 reconstructive surgery. *Biomechanics and Modeling in Mechanobiology*
25 **17**, 1857–1873 (2018)
- 26 [43] Virtanen, P., Gommers, R., Oliphant, T.E., Haberland, M., Reddy, T.,
27 Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van
28 der Walt, S.J., Brett, M., Wilson, J., Millman, K.J., Mayorov, N., Nelson,
29 A.R.J., Jones, E., Kern, R., Larson, E., Carey, C.J., Polat, İ., Feng,
30 Y., Moore, E.W., VanderPlas, J., Laxalde, D., Perktold, J., Cimrman,
31 R., Henriksen, I., Quintero, E.A., Harris, C.R., Archibald, A.M., Ribeiro,
32 A.H., Pedregosa, F., van Mulbregt, P., SciPy 1.0 Contributors: SciPy
33 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods* **17**, 261–272 (2020). <https://doi.org/10.1038/s41592-019-0686-2>
- 34 [44] Tepole, A.B., Gart, M., Gosain, A.K., Kuhl, E.: Characterization of living
35 skin using multi-view stereo and isogeometric analysis. *Acta biomaterialia*
36 **10**(11), 4822–4831 (2014)
- 37 [45] Ghaboussi, J., Pecknold, D.A., Zhang, M., Haj-Ali, R.M.: Auto-
38 progressive training of neural network constitutive models. *International Journal for Numerical Methods in Engineering* **42**(1), 105–126
39 (1998) [https://onlinelibrary.wiley.com/doi/abs/10.1002/\(SICI\)1097-0207\(19980515\)42:1<105::AID-NME356>3.0.CO;2-V](https://onlinelibrary.wiley.com/doi/abs/10.1002/(SICI)1097-0207(19980515)42:1<105::AID-NME356>3.0.CO;2-V)
- 40 [46] Ghaboussi, J., Sidarta, D.E.: New nested adaptive neural networks (nann)
41 for constitutive modeling. *Computers and Geotechnics* **22**(1), 29–52
42 (1998). [https://doi.org/10.1016/S0266-352X\(97\)00034-7](https://doi.org/10.1016/S0266-352X(97)00034-7)
- 43 [47] Kirchdoerfer, T., Ortiz, M.: Data-driven computational mechanics. *Computer
44 Methods in Applied Mechanics and Engineering* **304**, 81–101
45 (2016)
- 46 [48] Heider, Y., Wang, K., Sun, W.: So (3)-invariance of informed-graph-based
47 deep neural network for anisotropic elastoplastic materials. *Computer
48 Methods in Applied Mechanics and Engineering* **363**, 112875 (2020)
- 49 [49] Mihai, L.A., Woolley, T.E., Goriely, A.: Stochastic isotropic hyperelastic
50 materials: constitutive calibration and model selection. *Proceedings of the
51*

30 *Data-driven Soft Tissue Mechanics*14 Royal Society A **474**(2211) (2018)

- 16 [50] Dabiri, Y., Van der Velden, A., Sack, K.L., Choy, J.S., Kassab, G.S.,
17 Guccione, J.M.: Prediction of left ventricular mechanics using machine
18 learning. *Frontiers in Physics* **7**, 117 (2019). <https://doi.org/10.3389/fphy.2019.00117>
- 20 [51] Kalina, K.A., Linden, L., Brummund, J., Metsch, P., Kästner, M.: Automated
22 constitutive modeling of isotropic hyperelasticity based on artificial
23 neural networks. *Computational Mechanics* **69**(1), 213–232 (2022)
- 25 [52] Leshno, M., Lin, V.Y., Pinkus, A., Schocken, S.: Multilayer feedforward
26 networks with a nonpolynomial activation function can approximate any
27 function. *Neural networks* **6**(6), 861–867 (1993)
- 29 [53] Klein, D.K., Fernández, M., Martin, R.J., Neff, P., Weeger, O.: Polyconvex
30 anisotropic hyperelasticity with neural networks. *Journal of the Mechanics
31 and Physics of Solids* **159**, 104703 (2022)
- 33 [54] Mora-Macías, J., Ayensa-Jiménez, J., Reina-Romo, E., Doweidar, M.H.,
34 Domínguez, J., Doblaré, M., Sanz-Herrera, J.A.: A multiscale data-driven
35 approach for bone tissue biomechanics. *Computer Methods in Applied
36 Mechanics and Engineering* **368**, 113136 (2020)
- 38 [55] He, Q., Laurence, D.W., Lee, C.-H., Chen, J.-S.: Manifold learning based
39 data-driven modeling for soft biological tissues. *Journal of Biomechanics*
40 **117**, 110124 (2021)
- 42 [56] He, X., He, Q., Chen, J.-S.: Deep autoencoders for physics-constrained
43 data-driven nonlinear materials modeling. *Computer Methods in Applied
44 Mechanics and Engineering* **385**, 114034 (2021)
- 46 [57] Kassab, G.S., Sacks, M.S.: *Structure-based Mechanics of Tissues and
47 Organs*. Springer, ??? (2016)
- 49 [58] Kumaraswamy, N., Khatam, H., Reece, G.P., Fingeret, M.C., Markey,
50 M.K., Ravi-Chandar, K.: Mechanical response of human female breast
51 skin under uniaxial stretching. *Journal of the mechanical behavior of
52 biomedical materials* **74**, 164–175 (2017)
- 54 [59] Wang, Z., Estrada, J.B., Arruda, E.M., Garikipati, K.: Inference of deforma-
55 tion mechanisms and constitutive response of soft material surrogates
56 of biological tissue by full-field characterization and data-driven varia-
57 tional system identification. *Journal of the Mechanics and Physics of
58 Solids* **153**, 104474 (2021)
- 60 [60] Sivaloganathan, J., Spector, S.J.: On the uniqueness of energy minimizers

17 in finite elasticity. *Journal of Elasticity* **133**(1), 73–103 (2018)
18

- 19 [61] Bonet, J., Gil, A.J., Ortigosa, R.: A computational framework for poly-
20 convex large strain elasticity. *Computer Methods in Applied Mechanics and Engineering* **283**, 1061–1094 (2015)

- 21 [62] Tac, V., Costabal, F.S., Tepole, A.B.: Automatically polyconvex strain
22 energy functions using neural ordinary differential equations. arXiv
preprint arXiv:2110.03774 (2021)

- 23 [63] Baillargeon, B., Rebelo, N., Fox, D.D., Taylor, R.L., Kuhl, E.: The living
24 heart project: a robust and integrative simulator for human heart
25 function. *European Journal of Mechanics-A/Solids* **48**, 38–47 (2014)

- 26 [64] Cilla, M., Pérez-Rey, I., Martínez, M.A., Peña, E., Martínez, J.:
27 On the use of machine learning techniques for the mechanical
28 characterization of soft biological tissues. *International Journal for
29 Numerical Methods in Biomedical Engineering* **34**(10), 3121 (2018)
30 <https://onlinelibrary.wiley.com/doi/pdf/10.1002/cnm.3121>. <https://doi.org/10.1002/cnm.3121>. e3121 cmn.3121

- 31 [65] Wriggers, P., Korelc, J.: On enhanced strain methods for small and finite
32 deformations of solids. *Computational Mechanics* **18**(6), 413–428 (1996)

33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65



Click here to access/download
Supplementary Material
NNMAT_supplement1.pdf



Click here to access/download
Supplementary Material
Supplement 2.jpg